Tietojenkäsittelyliiton monistesarja

1/79

Summary report of the Systemeering research seminar of Tampere 21.–24.8.1978

SUMMARY REPORT OF THE SYSTEMEERING RESEARCH SEMINAR AT TAMPERE 21.-24.8.1978

EDITED BY

P.KEROLA
M.KLEMOLA
H.KAMARAINEN
K.LYYTINEN

PREFACE

The purpose of this report is to give information about problems and activities in systemeering research. The 'systemeering' is a special term used by Scandinavian scientists for systems analysis and design (of a data processing or information system) - hence the analogy with the term 'programming' in its general sense and in the specific connection of computers. Special attention is paid to infologically oriented research in which the objects of study are

- people and organizations utilizing information
- criteria for the significance and usability of the data processing system
- interactions between the data processing system and its environment
- all the processes and functions of the information system during its life-cycle, and especially
- systemeering processes.

The main objective is to balance the infological perspective with a datalogical one (computer and data oriented).

The group working on this topic has so far been quite informally organized, being based mainly on personal contacts and co-operation in similar research problems. This common effort was started by a 'change analysis' of systemeering research work and the results are reported in the second chapter. The main idea was to organize a systemeering research seminar in which the special topics would be the problems of research work and the terminology to be used. It was decided that all the participants should prepare preliminary papers as a basis for the group work (results in the Chapters II and III). After the seminar, the participants then had an opportunity to submit comment papers on the preliminary material, discussion and accounts of experiences during the seminar (Chapter IV). Some future plans for joint work are presented in the last chapter.

This summary report has been prepared "with the minimum resources necessary to achieve a satisfactory level", to distribute information on the activities concerned. The technical quality is not the best possible, but it is hoped that this will not disturb the reader.

The participants appreciate very much financial support for the seminar received from the Research Committee of the Finnish Data Processing Association.

Oulu 31.12.1978

On behalf of the editors

Pentti Kerola
Professor
Institute of Data Processing Science
University of Oulu

LIST OF PARTICIPANTS

Göran Goldkuhl, univ. lector	University of Stockholm
Mats Lundeberg, docent	- " -
Pentti Kerola, prof.	University of Oulu
Seppo Mustonen, assist.prof.(acting)	- " -
Kari Eloranta, docent	University of Tampere
Pertti Järvinen, assoc.prof.	- " -
Hannu Kangassalo, res.assistant	- " -
Heikki Kämäräinen, assistant	- " -
Pekka Tyllilä, res.assistant	- " -
Mikko Klemola, project leader	University of Jyväskylä
Kalle Lyytinen, assistant	- " -
Eero Peltola, prof.	- " -
Timo Järvi, prof.	University of Turku
Markku Nurminen, univ.lector	School of Economics, Turku

TABLE OF CONTENT

I PURPOSE OF SEMINAR 1

Program 1

II PRELIMINARY ACTIVITIES 2

1. Change analysis reports 3

2. Preliminary papers .....

List of published papers 9

Goldkuhl-Lundeberg: A note on ISAC's research philosophy 10

Kangassalo: Conceptual foundations for data base management terminology 12

Kerola-Järvinen: PSC-systemeering model and its influence on basic concept structure of data system development 30

Klemola-Lyytinen: On the conceptual foundations of systemeering - a semiotic approach 39

Kämäräinen: An analysis of data processing system and the activities related to it 66

Mustonen: On cybernetic analysis as a basis of metatheories in systemeering 110

Nurminen: Properties and problems of a general-purpose data system 126

Peltola-Maanavilja-Laine: Grammatical data base model, I Description of reality by expressions of a formal language 136

III SUMMARY OF GROUP WORK 160

1. On the problems of performing research work in systemeering (WG1) 161

2. Definition of some basic concepts (WG2) 171

IV COMMENT PAPERS 187

T.Järvi: On WG2 report 188

P.Kerola: Notes on terminology, abstraction levels and levels of detail in PSC model of systemeering 189

H.Kämäräinen: Notes on the PSC model of systemeering 194

M.Lundeberg-G.Goldkuhl: Condensed comments 209

K.Lyytinen: How to define data system and information system 216

P.Tyllilä: Notes concerning levels of detail and levels of abstraction 223

V FUTURE PLANS 237

I PURPOSE OF SEMINAR

To analyze basic problems and concepts in the research of systemeering and to discuss about the research approaches to solve those problems.

As a final outcome the English written report of the seminar will be published.

Program

21.8. 11.00 Lunch
12.00 Opening
12.30 I Introductory and analysis phase
- introductory presentations and discussions à 1.5 hours
1. Stockholm University
2. Oulu -"-
3. Tampere -"-
4. Jyväskylä -"-
22.8. 5. Turku -"- and School of Ec.
6. Others

12.30 II Science Theory and Methodology in relation to Information Processing Science
- presentations
- discussions

23.8. III Synthesis phase
- group work
- discussions

24.8. IV Summary
- plan for future meetings

12.00 Lunch
To home

1. Every participant (alone or together with somebody) will prepare and/or distribute (latest at the end of July) the English or Swedish written paper(s) about his (their) research philosophy in systemeering and basic set of concepts and definitions

2. Everybody has thoroughly studied the papers of the others before the seminar and is ready to discuss, analyze and make synthesis about the subject concerned.

3. Introductory presentations will be short and the purpose of those is just to open discussions (and mind)

II PRELIMINARY ACTIVITIES

1. CHANGE ANALYSIS REPORTS

sac
ats Lundeberg
öran Goldkuhl

ANALYSIS OF PROBLEMS AND CURRENT SITUATION REGARDING RESEARCH IN THE SYSTEMEERING AREA

F11 LISTING OF PROBLEMS

P1 Generally accepted theories that are possible to communicate are lacking in the area. Many researchers are working with their own things from their own perspectives. There is a tremendous unused potential here. It should be possible to use this potential if one could use each others' results more.

P2 A thoroughly worked through theory and methodology of science is lacking for the systemeering area. Such a theory must take the following sciences into account: Social, behavioral, logical and technical.

P3 Some research in the area is too much in the "model" world with weak links to reality. This may be due to the circumstance that researchers have not thought about what theories should be used for and in what connection.

P4 Severe antagonism exists between problem oriented and data oriented systemeering researchers because of different perspectives.

P5 There also exist different perspectives within these respective areas.

P6 It appears that the academical system hinders work in groups, too much individual work dominates at present.

P7 There is a lack of inspiring, creative and supporting research environments. The research resources in the area are scarce.

F12 ANALYSIS OF INTEREST GROUPS

We have found the following interest groups:

I1 Researchers (end users)

I2 Practitioners (general public)

I3 Researchers, practitioners, educators, politicians (funders)

I4 Theoreticians of Science (specialists)

I5 Students (future researchers and practitioners)

Pentti Kerola-Seppo Mustonen
Institute of Data Processing Science
University of Oulu

CHANGE ANALYSIS FOR THE SUBJECT AREA:

BASIC PROBLEMS AND CONCEPTS OF SYSTEMEERING RESEARCH

A List of problems

1 It doesn't exist any generally accepted set of basic concepts - absolutely and necessarily needed for effective progress in the development of systemeering theory.

2 Systemeering theory must be based on the theoretical results of philosophy, systems theory, natural, social and technical sciences - there are very few if anybody whose higher education covers many of those classical branches of the science. Therefore well-balanced co-work is absolutely needed between the groups of researchers.

3 The identity of computer science, data (information) processing science, informatics - obs. point 2.

4 Youth of 'data processing society'- there has been little time for the development of theory.

5 Conflicts between the infologically and datalogically oriented researchers and practitioners.

B Interest groups and their problems

1 Scientists, teachers and students in educational and research institutions (A1,A2,A5)

2 People of practice in DP area
- managers (A3,A4)
- end users of data system outputs (A1,A4,A5)
- staff specialists (not DP) (A1,A3,A4,A5)
- DP professionals
  - managers (A4,A5)
  - systemeerers (all)
  - programmers (A4,A5)
  - operational

3 Organizations of standardization (A1,A4,A5)

4 DP organizations (A1,A3,A4)

C Problem grouping

1 Research ideologies and their coordination

2 Terminology and definitions of concepts

3 Human conflicts

4 Applicational problems

D Plan for change

1 Possibilities for change

1 There are very many ways dependant on
- the amount of recources
- how rapidly some positive results are needed
- how many and what kind of people are considered

2 Different possibilities can be analyzed but now out of question - so we seek just a small step with most probably positive results during the next year.

2 Purpose

To propose the next step for the restricted and selected interest group mainly from the Swedish and Finnish universities in the development of basic concepts of systemeering with
- small financial resources
- positive and open attitude for cowork
- deep interest in development of systemeering theory

The first and third problem group (C1,C3) are now at least reasonably diminished by the selection of participants. The main problem is that the infological researchers have basically better approach for the development of concepts in systemeering but the relative quality of their definitions is lower than datalogical researchers have. On the other side they "don't even see" the whole set of object systems in systemeering, they are more restricted.

3 Special goal

Condensed English written manuscript for the scientific paper which later can be finally worked and published.

4 Content

4-5 days meeting in Tampere at August-September 1978 with the following necessary activities of every participant:

- to prepare and distribute English or Swedish written paper about his
  - research ideology
  - basic terms, concepts and their definitions
  - main results for cowork progress (own proposal)
- to read the main written material of other participants regarding 'normal' language restrictions
- to endure 'social possibilities' with small financial resources

At least two days are needed for the presentations (analysis phase) and the same for synthesis phase.

2. PRELIMINARY PAPERS

List of earlier published papers

/1/ Järvi: A proposal of an extension of information algebra
ANN. UNIV. TURKUENSIS SER. A.1.175 1977

/2/ Järvinen-Kerola: Notes on research in systemeering
ANN. UNIV. TURKUENSIS SER. A.1.175 1977

/3/ Järvinen: Notes on educational planning: A systems approach
ACM SIGCSE BULL. Dec 1977

/4/ Kerola: On hierarchical information and data systems in data systems life cycle. Systemering 75, Studentlitteratur 1975

/5/ Lundeberg, M., Goldkuhl, G., Nilsson, A.: Information systems development - A first introduction to a systematic approach (English draft available from research group ISAC, Administrative information processing, University of Stockholm, S-106 91 Stockholm, Sweden); 1978

isac
Göran Goldkuhl
Mats Lundeberg

A NOTE ON ISAC'S RESEARCH PHILOSOPHY

The ISAC group has performed research around systemeering methodology since 1971 with an emphasis on problem oriented systemeering. We have in this work interacted between development, application and evaluation of methodology. The empirical part (methods application) has mainly consisted of participation in real systemeering projects. In these projects ISAC methodology has been used and we have participated as ISAC researchers. Our task has been to work as educaters, "additions of systems analysts", method supporters, catalysts and observers (i.e. collectors of empirical material). This work has thus been a kind of action research. Such research is characterized by such things that the researcher acts in his observation field and that the cycles between empirical work ("action part") and theoretical work ("reflection part") are rapid and occur often.

Why have we been working with action research? Why have we not used more usual investigation strategies such as surveys or laboratory experiments? We have used such strategies as well, but to a limited extent. We think that we have identified a larger number of problems in the "reality of systemeering" through action research. We have come closer to the problems of reality than we think we would have with more traditional methods. We think closeness to the problems is essential in a young field. It is important to define relevant empirical questions "exploratively". We have been able to study certain variables more carefully through action research than with e.g. survey approaches. It is possible that we thereby lose in generality although we ourselves are not so sure about that.

We have made different preparations before the applications, e.g. formulation of hypotheses. We have evaluated methods applications by preparing collected empirical material

(project documentation from the systemeering work and documented observations from the systemeering process in logg books). We have analyzed and documented this material in reports about our experiences. We have tested certain hypotheses against the empirical material and also generated new hypotheses. We have also done some work in the area of putting together hypotheses in systems of hypotheses (theories).

We consider our area of study - systemeering - as a cross scientific field between different knowledge areas: Technology (data processing), logics, knowledge theory, semiotics and social sciences (primarily behavioral and organizational sciences). In this cross scientific field it is important to have a suitable view and approach for different subproblems. It is important not to take views from the technical area to the human area and vice verse. What is suitable for computers (deterministically controlled machines) is often not suitable for human beings (conscious actors that give meaning to themselves and others in a social context).

CONCEPTUAL FOUNDATIONS FOR DATA BASE MANAGEMENT TERMINOLOGY

October 1978.

Hannu Kangassalo

University of Tampere
P.O.Box 607
SF-33101 Tampere 10
Finland

For developing a consistent terminology for data base management field a sound conceptual foundation is needed. The purpose of this paper is to analyse the concept of data, other concepts associated with the concept of data as well as associations between them, and to make a proposal for some basic concepts for data base management.

This paper is a completely rewritten version of the earlier paper 'DATAMANAGEMENT TERMINOLOGY' of the Data Management -group of the Finnish Data Processing Association. The present paper lacks the concepts concerning a database approach and a data administration function which were included in the earlier paper

This paper has been distributed as the Finnish contribution for discussion at the ISO/TC97/SC5/WG3 DBMS-meeting in Helsinki 1978-10-17...20.

1. Introduction

In the data management field there is a widespread opinion nowadays that data stored in a data base are part of a model representing some part of the real world. However, there is no generally accepted view concerning the relation between the real world and data incorporated in the data base. The description of this relation describes the semantical 'meaning' of data. The reason of this disagreement is the fact that there are several different conceptions of 'meaning' as well as different philosophical disciplines which these conceptions are based on. Therefore it may be the case that further elaboration in the field of the theory of meaning is needed before we can choose the set of basic concepts for data base management.

The purpose of this paper is to analyze the concept of data, other concepts associated with the concept of data as well as associations between them, and to make a proposal of the set of basic concepts in the field of data base management.

2. Analysis of basic concepts

The real world and the phenomena within it are described using data. Also abstract, illusory things can be described as data. However, data is not directly related with real world. Three 'realms' or 'levels' can be identified: the linguistic, the conceptual and the physical level /1, p. 57/.

The linguistic level is concerned with the language and components of the language such as proper names, terms, phrases and sentences etc.

The conceptual level is concerned with concepts and conceptual constructs, such as propositions, theories, conceptual models etc. Although it may be impossible to find a complete definition of the notion of 'concept', we say that a concept is a pattern or a logical construct created by human thinking. In this paper we think that concepts exist in the same sense as numbers. We can also say that concepts are abstract objects which are materialized

in human brain processes. By using concepts people tend to outline, organize or analyze the abundance of the things and the phenomena observed in the real world. The physical level consists of the set of all physical entities and of the set of all phenomena within the set of physical entities, such as inserting a new entity, change of properties of an entity, and all kinds of physical process. The discerning of the linguistic level and the conceptual level from the physical level is a purely theoretical distinction because both of these levels can be analyzed also from th physical point of view, and in fact they are part of the physical reality. Their special status is based on the functions of objects on these levels and on the relations between the objects on these levels and other parts of physical reality. Figure 1 depicts these levels, some objects on these levels, and relations between these levels / 1, p. 58/.

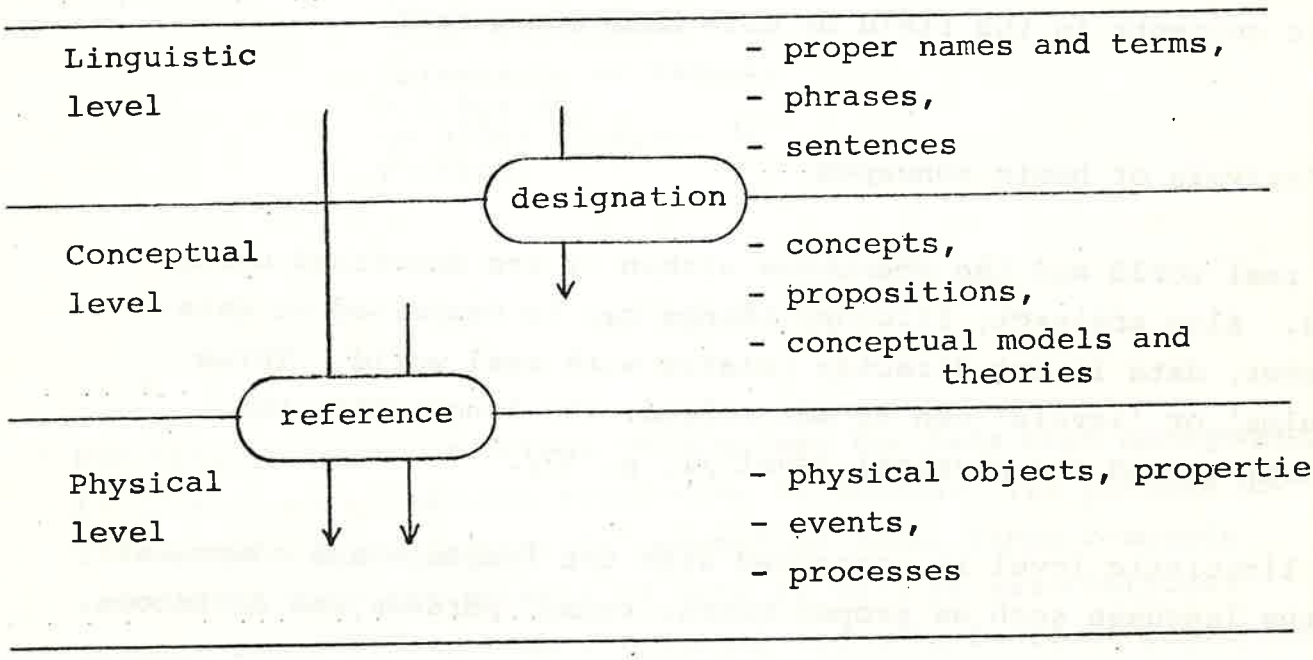


Figure 1. Different levels, their contents and relations between them.

The relations between different levels are the relation of designation and the relation of reference. The relation of designation holds between some objects of the linguistic level

and their correlates, if any, on the conceptual level. The relation is an asymmetrical sign-idea relation. In general it is not a one-to-one relation. The relation of reference holds between some members of either the linguistic or the conceptual level and their correlates, if any, in the physical level. The relation of reference is an asymmetrical relation.

If a proper name (on the linguistic level) or a concept (on the conceptual level) has a real correlate in the physical level, we shall say that the proper name or the concept refers to that entity or thing. A term is the name of a concept. It is a linguistic entity which can be used to designate the concept. The union of the two relations, that of designation and that of reference, may be called denotation / 1, p. 58/.

Concepts are described using definitions. The definition is a linguistic expression, which associates a new concept - the definiendum - with the set of old, i.e. already known, concepts - the definiens - in some specific way. The definition describes the properties of a new concept with the help of properties of old concepts. However, within the system of concepts a set of undefined, intuitively understandable basic concepts is always needed.

In the following the term object is used as a general term to denote anything which is under observation. Let O designate a set of objects o_i, $i=1,2, \ldots ,n$. Every object has a set of properties P_i which characterizes that object. The type of an object o_i is defined by those of its properties common to all elements in O together with the set operations which are defined in O and which are applicable to elements of O.

Let C designate the set of all concepts and let V designate a subset C, i.e. $V \subset C$. The elements of V, which are called values, are all of the same type. The value v, $v \in V$, is a concept which is selected from a given set V by some operation t, e.g. measurement, classification or calculation. The type of a value v can be, for example, numerical or logical, but it is not limited to these two possibilities only.

The definition of a value, which describes the properties of this value in relation to the system of other values of the same type, is usually implicitly contained in some branch of the mathematical or logical theory. The definition of a value may also contain a set of auxiliary rules, which are not part of that theory, but in a specific context express certain properties of the set V, e.g. the maximum and the minimum value of the set.

Because the value itself is a concept it must be possible to expre values by some means on the linguistic level. A representation of a value is a linguistic expression which with the help of the set of physical signs materializes the value according to the represen tation rules of a given representation language. The representati language or the notation is any generally known or agreed language which is suitable for representing values.

However, in some situations it is necessary to make a distinction between the type of a representation and a token of a representati The type of a representation of a given value is the totality of properties of that representation defined by the set of represen- tation rules in a given representation language together with the value to be represented. The token of a representation of a give value is defined by the totality of physical properties of that representation which are determined either by the set of represen tation rules or by the instantiation process which locates the representation at a particular place in space and in time.

For example, the type of a representation of a given value determ nates characters to be used in representation and the sequence of these characters but the type does not specify the actual locatio of the representation. The token of that representation consists of a sequence of actual physical characters which are located at a specific place at a certain moment.

The description of the notation is a set of rules which describes the syntactical structure of the representation and gives semant rules for the interpretation of the representation. The descrip

of the notation contains two different parts: a metalanguage defines the whole representation language, an instance description describes (or prescribes) those properties of a token of a representation which are determined by the instantiation process at the moment of materialization of a representation.

The relations between these concepts can be depicted as in figure 2. In figure 2 the relations between concepts are described by arrows. There is a verb attached to every relation describing the type of a relation. The descriptions can be read as like the following example: 'A TOKEN OF THE REPRESENTATION instantiates the TYPE OF THE REPRESENTATION'.

In figure 2 there is only one instance of 'TERM' because in the data processing field usually one and the same term is used to name a concept, e.g. 'salary', a value associated with that concept, a type of the representation and a token of the representation. However, it is quite possible to use a separate name for all these objects if necessary.

Figure 2 describes the situation when only one representation, one value and one concept are involved. If several representations, values and concepts are involved the situation becomes more complicated because of two reasons. Firstly, for each additional representation, value and concept, one additional description is needed. Secondly, if the concepts and/or the values are somehow related with each other, the descriptions of the relations are needed. Usually these descriptions are incorporated in the concept definitions and in the value definitions, but in some cases they can also be separate objects. We shall not analyze the latter case here in more detail.

On the conceptual level the real world is described using fact constructs. A fact construct is a conceptual construct which is either a composition of different concepts or a pair of a concept and the value associated with it. The fact construct of the former type relates two or more concepts with the help of one or more additional concepts. This type of fact constructs are called CR-facts (concept-relation-fact). Figure 3 shows an example of a CR-fact.

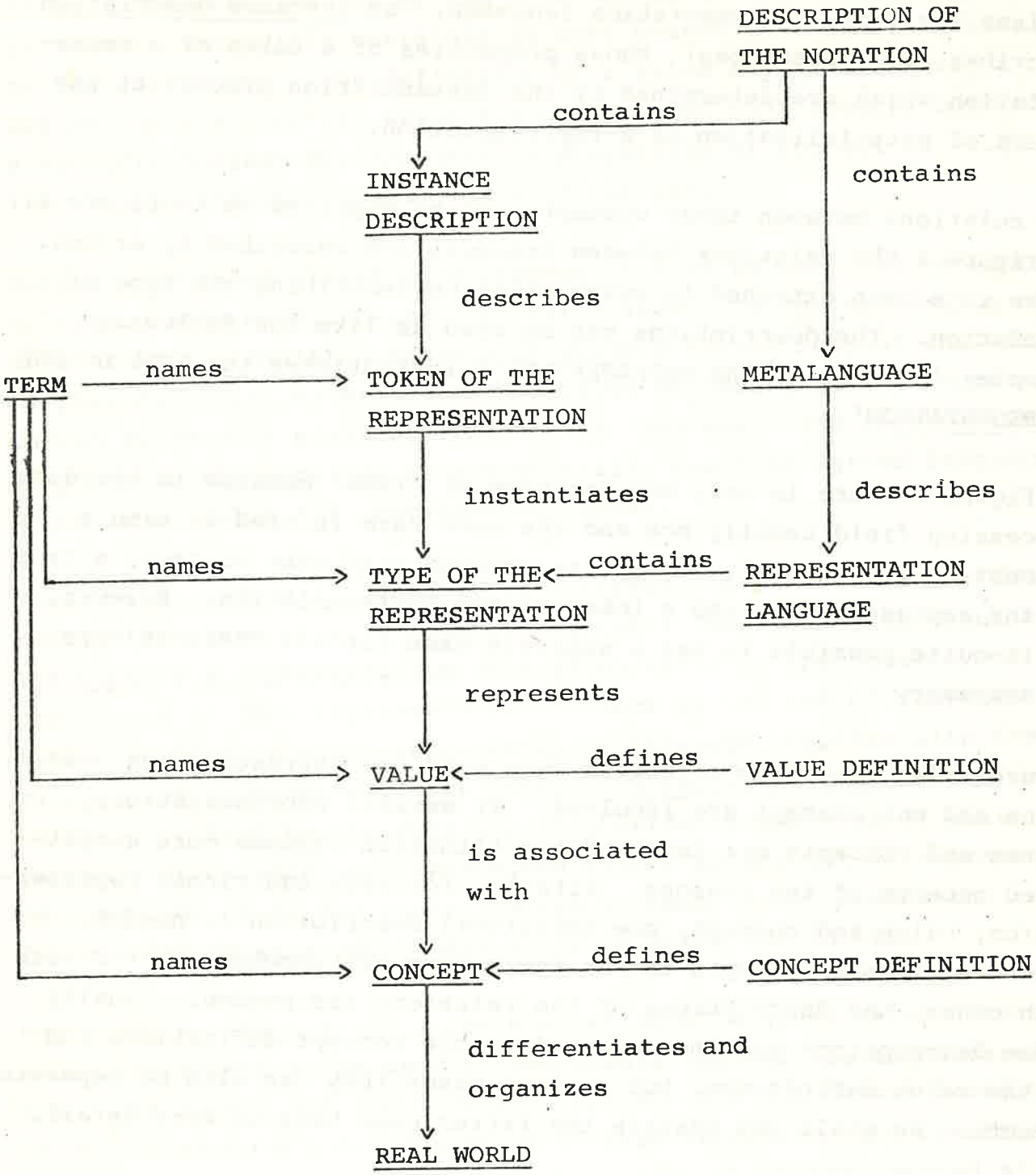


Figure 2. The set of basic concepts and relations between these concepts.

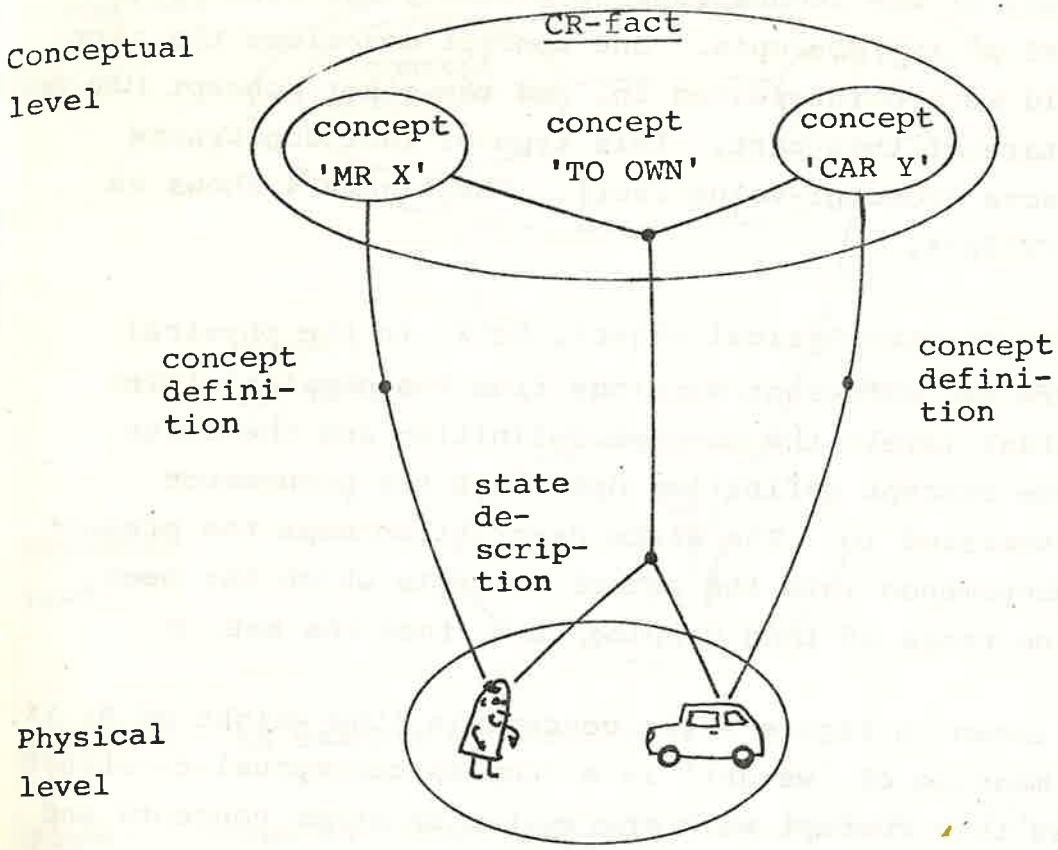


Figure 3. An example of the CR-fact.

In figure 3 there are two physical objects in the physical level (Mr X and car Y) and a coupling of certain type between these objects. The coupling consists of a set of different actions which Mr X has a right to perform with the car X.

There are four different mappings from the physical level into the conceptual level. Three of these mappings are concept definitions which define the concept of 'Mr X', the concept of 'car Y' and the concept of 'to own'. The definition of 'Mr X' and the definition of 'car Y' describe the properties of these objects. The definition of 'to own' describes the set of different actions which the owner has a right to perform with the object which is owned. This definition has not been shown in the figure 3.

The fourth mapping is a state description function which maps (a part of) the state of the real world into the set of conceptual constructs. In this example the concept 'to own' links the concepts 'Mr X' and 'car Y' together.

The fact-construct of the second type is a conceptual construct which is composed of two concepts. One concept describes the part of the real world we are interested in, and the other concept (the v describes the state of this part. This type of fact constructs are called CV-facts (concept-value-fact). The figure 4 shows an example of the CV-fact.

In figure 4 there is one physical object, Mr X, in the physical level. There are two different mappings from the physical level into the conceptual level: the concept definition and the state description. The concept definition describes the phenomenon which we are interested in. The state description maps the presen status of the phenomenon into the set of concepts which has been defined to be the range of this mapping, i.e. into the set V.

In the example shown in figure 4 the concept is 'the weight of Mr The conceptual meaning of 'weight' is a complex conceptual constru which associates this concept with the system of other concepts an relations between these concepts. The value is '75'. It is the result of a complex operation, which contains weighing Mr X with a scale and translating the analogue result into a digital form according to a set set of specified rules. The method of weighing and the set of translation rules connect the value with the comple conceptual system which defines the units of weighing, e.g. grams, kilos, etc.

On the linguistic level each fact-construct is represented as a fact-sentence. The fact sentence is a linguistic expression which represents the conceptual contents and the structure of the corresponding fact construct. For each concept in the fact consti there is a term in the fact sentence. The structure and syntactic form of the fact sentence corresponds to the structure of the fact construct according to the rules of the representation language.

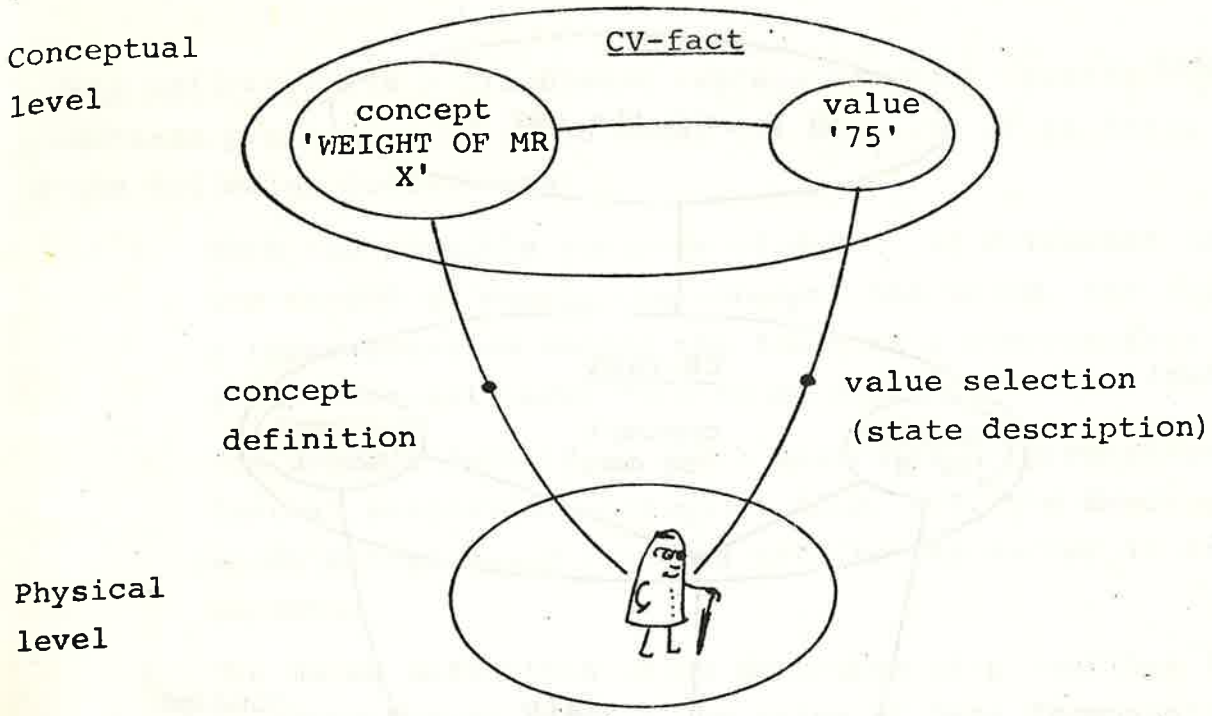


Figure 4. An example of the CV-fact.

Figure 5 shows an example of relations between different levels. The contents of the conceptual level and of the physical level is that shown in figure 3.

In figure 5 the statement 'Mr X owns the car Y' belongs to a given representation language and the interpretation rules of this language are generally known.

The reason for the separation of the fact construct and the fact sentence is that one fact construct can be represented with several different fact sentences which all are part of the same representation language. It can even be represented in several different ways in several different representation languages.

The concept of data can be defined using concepts defined above. Data is the construct which is composed of a fact construct and a representation of that fact construct. Both of these components are necessary. A representation without a corresponding fact construct is only a character string without any specific meaning.

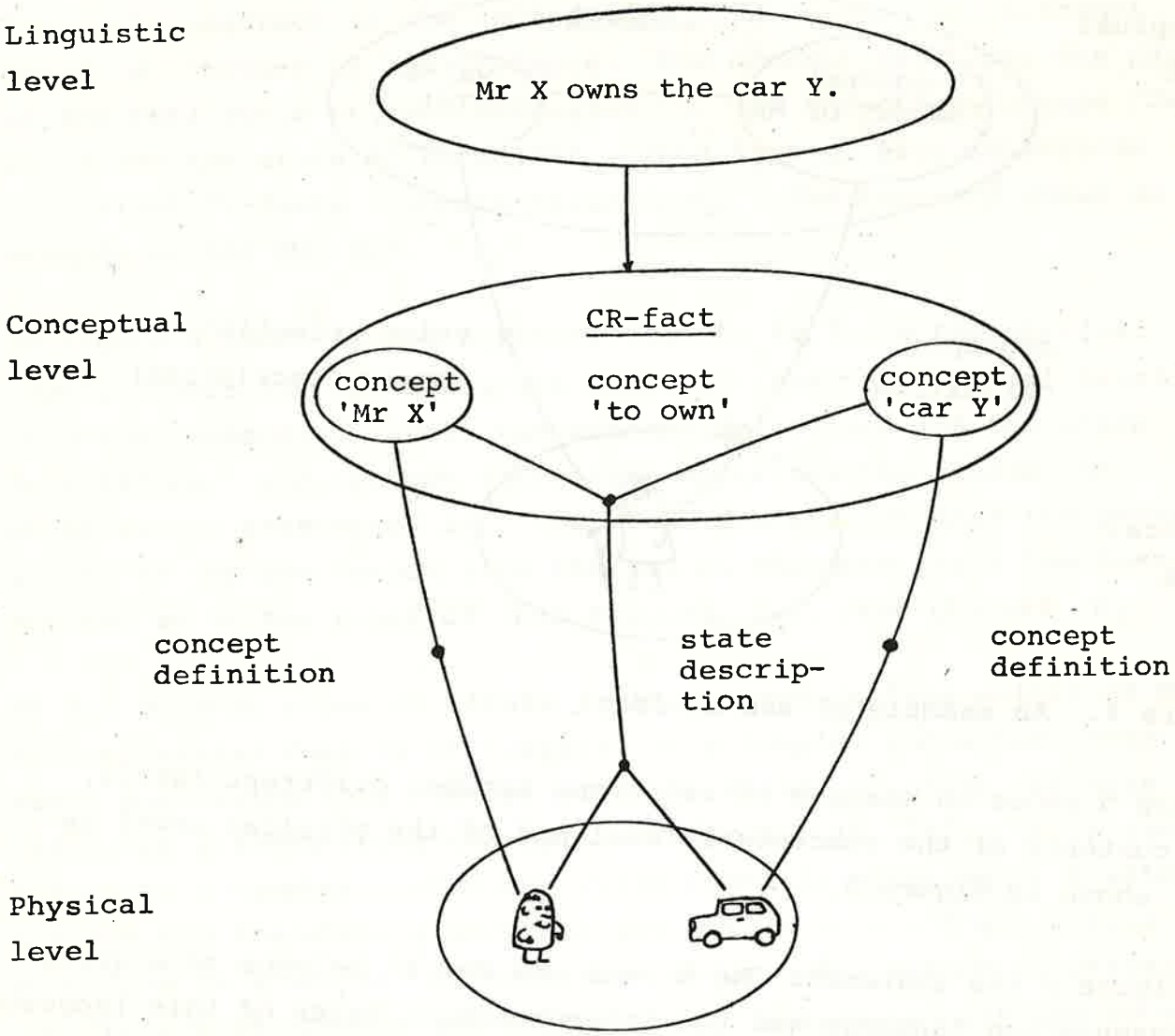


Figure 5. Three different levels.

On the other hand, it is impossible to express a fact construct without a representation.

This conception of data has some consequences to the documentation methodology. All four aspects of data described above, the conceptual aspect, the value aspect, the type of the representation aspect and the token aspect, have to be documented. The documentation method and the tools used in documentation must be able to manipulate all these aspects independently as well as some combinations of these aspects.

A data definition is a linguistic expression which describes or prescribes properties of given data. It consists of at least some of the following definitions:

1. Term and possible synonyms of data. If different terms are needed to denote the concept, the value, the type of a representation and/or the token of a representation of data, then all terms have to be defined.
2. The concept definition which describes or prescribes the factual semantical meaning of data, i.e. the description which conceptually connects data to the matter it describes.
3. The value definition which describes or p escribes the operation which selects the value of data from a given range of values.
4. The description of the type of a data representation.
5. The instance description of the token of a data representation.
6. A set of rules to describe or to prescribe the use of data.

A data item is the smallest piece of data as seen from the certain point of view. For example, date could be seen to consist of day 15, month 12 and year 1327 which are data items from some point of view. However, from some other point of view one could divide the year into centuries and decades and now these are data items. In most situations the term 'data item' can be used to designate both the type of a representation and the token of a representation without any ambiguity.

Usually data items are combined to form larger constructs of data. A data group is a collection of one or more data items and/or data groups. In the example above, date and year are data groups.

3. Data base management terminology

On the basis of the analysis presented in the previous chapter it is possible to define a set of complex conceptual constructs, which is the core of the conceptual structure of the data base management field.

The development of this set of concepts will proceed according to the principle of abstraction levels. The principle of abstraction levels means that a collection of data can be described with the hierarchically arranged set of models in which each model has been constructed using such a set of modelli concepts that within each set the modelling concepts are close to each other in their abstractness, but the sets differ from each other in the abstractness of their modelling concepts. The hierarchy of models is arranged in such a way that the most concrete modelling concepts are on the lowest level of the hierarchy and the most abstract modelling concepts are on the highest level of the hierarchy.

In this context modelling concepts are properties of data and/or relations between these properties. The properties and the relations between them can the classified into four different classes which are called abstraction levels. Each class of concepts is a set of modelling concepts on one level in the hierarchy of models. On each abstraction level there are some basic modelling concepts, from which it is possible to derive a set of other modelling concepts which belong to this abstractic level. The basic modelling concepts are (from the most abstract to the most concrete):

1. On the highest level
   - concepts and
   - associations between concepts.

2. On the second level
   - values and
   - associations between values.

3. On the third level
   - properties of data defined on the basis of representation rules of a given representation language and the values to be represented and
   - types of representations of associations between data entities.

4. On the lowest level
   - physical properties of tokens of data
   - properties of hardware used store and manipulate data.

On the lowest level the set of hardware properties cannot be abstracted from the set of data. However, the set of hardware properties have an effect on possibilities to represent and organize data. Therefore this set has to be taken into consideration.

The association is used in this paper as an undefined basic notion to express that there is a coupling between two concepts, i.e. that two concepts are connected with each other to form a more complex conceptual construct. It is important to notice that there are also associations between concepts representing entities and concepts representing relations, as in the fact construct described in figure 3, where concepts 'Mr X' and 'to own' are associated with each other.

The association describes the existence of a coupling between concepts rather than the transformation of concepts. The type of association is described with the set of specified rules, which depending on the case may or may not correspond to the definition of some concept. In the fact construct described in figure 3 the state description function defines that there is an association between the concept 'Mr X' and the concept 'car Y'. The type of this association can be described with the same set of rules which describes the meaning of the concept 'to own'. It is easy to see that the definition of a certain association depends on the level of details which we are using.

On the conceptual level there are two sets of concepts: the set of concepts which are used as values and the set of concepts which are not used as values. In the data base management field this distinction is important because it may be the case that in the data base there are several fact constructs which contain the same concept(s) but different state descriptions (i.e. values or relations between concepts).

The real world can be described using the conceptual equipment described above. There are two separate descriptions. One description is a conceptual model of the real world, the other is a linguistic representation of the conceptual model.

The conceptual model is a conceptual construct which is composed of concepts and of fact constructs and which describes the contents, the structure and the state of the real world as seen by the community of the users of that conceptual model. The conceptual model contains concepts which describe the contents and the structure of the real world, and values which describe the state of the real world. If C_M is the set of concepts, excluding values, in the conceptual model M, then the conceptual structure S_M of the conceptual model M is the subset of the cartesian product $\underset{i=1}{\overset{i=n}{X}} C_{M_i}$, i.e.

$$S_M \subseteq \underset{i=1}{\overset{i=n}{X}} C_{M_i} .$$

The conceptual structure describes the existence of associations of different types between concepts.

If we exclude from the conceptual model all values, we will get a conceptual schema. A conceptual schema is a conceptual construct which is composed of concepts and which describes the contents and the structure of the real world as seen by the community of the users of that conceptual schema.

The conceptual level can be divided into two sublevels: the concept level and the value level. The concept level is that part of the conceptual level which contains the conceptual schema. The value level is that part of the conceptual level which contains the values corresponding to the concepts in the conceptual schema.

On the linguistic level a representation of a conceptual construct and a description of a conceptual construct must be separated from each other. The linguistic description of a conceptual schema is called a schema description. For one conceptual schema there can be several different schema descriptions. The linguistic description of a conceptual model is called a model description. For one conceptual model there can be several different model descriptions. The linguistic representation of a conceptual model is called a data base.

In the data base environment the description constructed from fact constructs alone is not very useful. Fact constructs and values have to be grouped together to form larger constructs in such a way that different data requirements can be met as efficiently as possible.

For processing purposes there has to be one or more access paths defined within the set of fact constructs. The access path element is a specified association between two values. Depending on the case the association may be either directed or undirected. The access path element may be defined in such a way that it corresponds to some concept or it may be defined to connect, without any specific meaning, two distinct values. The access path for the specified access requirement within the set of fact constructs is a sequence of access path elements, which is constructed in such a way that by proceeding along that sequence it is possible to reach all values which are needed in order to satisfy the access requirement.

Because the access requirement can be described as a hierarchy of lower level access requirements, it is also possible to construct hierarchies of access paths. These hierarchies of access paths are called abstract data structures. The abstract data structure is a construct which is composed of the set of fact constructs and of the (possibly hierarchical) access path which is defined in that set. The abstract data structure defines possible sequences in which the process can proceed within the set of fact construct.

When the type of representation of values and the implementation methods of access path elements of an abstract data structure have been defined, the construct is called a data structure. The data structure has the same logical properties than the original abstract data structure plus some additional logical properties which are inherited from the implementation methods of access path elements. Some of its physical properties have been defined but it has not yet been materialized by mapping it onto the data medium. The set of storage allocation rules, which are applied in mapping a data structure into the physical storage, defines the storage structure of a data structure.

The storage structure of a data structure is a set of relations between data items, between groups of data items and between data items and groups of data items stored in the physical storage.

REFERENCES

1. Bunge, M.: Scientific Research I. The Search for System. Springer-Verlag, Berlin 1967.

2. Lyons, J.: Semantics 1. Cambridge University Press, Cambridge 1977.

3. Kangassalo, H.: On defining and satisfying information requirements. (in Finnish). A thesis for the degree of licentiate in philosophy. University of Tampere, 1978.

4. Christensen, K.: Datamanagement terminology. An internal working paper of the Data Base Management-group, of the Finnish Data Processing Association, 22.3.1978.

5. Kangassalo, H.: Stratified description as a framework of the file description. (in Finnish). DIFO-vuosikirja 2/1973. DIFO-tutkimus ry. Tampere 1973.

6. Kangassalo, H.: Conceptual structure of the file design problem: a general view. Scandinavian Workshop on Data Base Schema Design and Evaluation, Røros, Norway, February 1975.

Pentti Kerola - Pertti Järvinen

12.7.78
DRAFTING PAPER FOR
"INTERNAL" RESEARCH
SEMINAR ON SYSTEMEERING
21.-24.8.78 IN TAMPERE

THE PSC SYSTEMEERING MODEL AND ITS INFLUENCE ON THE BASIC CONCEPT STRUCTURE OF DATA SYSTEM DEVELOPMENT

I INTRODUCTION AND SUMMARY

II ON CONSTRUCTS OF THE PSC SYSTEMEERING MODEL

III ON THE HIERARCHICAL CONCEPT STRUCTURE OF SYSTEMS DEVELOPMENT[1)]

- - - -

1) This chapter and the appendices connected to it have been omitted because of their introductory nature and because the seminar was too short to handle the material.

I INTRODUCTION AND SUMMARY

This is intended to be an introductory drafting paper on the research into systemeering going on in the universities of Oulu, Jyväskylä and Tampere under the quidance of the authors. The purpose is to draft a basis for the internal research seminar on systemeering. Most of the results have been published previously in Finnish.[1)]

The background to the research is an empirical one formulated during practical systems work in 1960-1973. The main objective has been normative: namely, to determine how to formulate a systemeering model in which the infological and datalogical aspects are balanced one with another. The scientific basis of research work has been multidisciplinary, utilizing the results of general systems theory, semiotics, cybernetics, information theory, economics and the social and behavioral sciences.[2)]

The main results are expressed as a set of constructs, by means of which we have developed the PSC systemeering model and some important concepts concerning it:

- an (ideal-seeking) purposeful system and its necessary components
- main functions of a purposeful system
- main aspects (perspectives) in systems analysis
- partition principles for purposeful systems
- stratified description of a system
- content of the concept 'systemeering'

Some of these results have been published in English (publ.2-4), and a great deal of research work has been done on evaluating the PSC systemeering model over the last two years.

- - - -

1) The selection of publishing channels has been largely governed by pedagogical considerations.

2) Some hypotheses are not yet formalized accurately enough, but are intuitively well-defined.

II ON CONSTRUCTS ON THE PSC SYSTEMEERING MODEL

In publ 4 we give a general description of the PSC systemeering model. Our effort here will be concentrated solely on the reasoning behind certain basic constructs to be used in the formulation of the model. In the following we present the pedagogically meaningful way of constructing the PSC systemeering model. The final result is the most important, and the intermediate ones only marginally so.

Data system as a whole

The first construct to be employed in the PSC systemeering model is the set of three aspects (publ. 2): the P aspect, the S aspect and the C aspect. These focus upon the data system under development. Crudely speaking, the application of these aspects to a data system would answer to the pairs of questions WHY-how, WHAT-how and HOW-how, corresponding to the P, S and C aspects respectively. The first part in each question pair states a view on the system after the application of the given aspect, while the second part describes the design results concerning that aspect.

The second construct is based on partition of the control function. The simplest partition will be between design (D) and test (T). By combining these two constructs we get the hierarchical PSC structure (Fig 1)

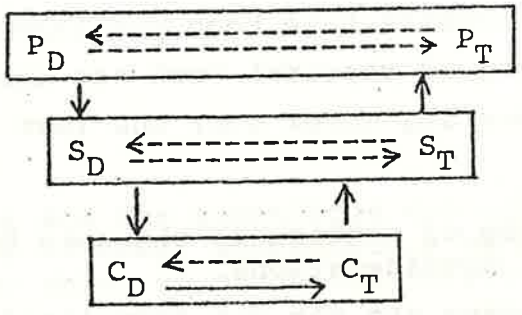


Fig 1. The hierarchical PSC-structure

The continuous lines in Fig 1 refer to the sequence of performance in a succesful case, and the dotted lines to feedback loops.

The hierarchical PSC structures can be presented in sequence form in the following way:

(I) $P_D S_D C_D C_T S_T P_T$

According to the sequence I , we have the following tasks:

P_D = pragmatic design
S_D = semantic design
C_D = construction design
C_T = construction test
S_T = semantic test
P_T = pragmatic test

in this order. The testing tasks will be performed in the opposite order from the design task, for the natural reason that it is not sensible to test the semantic features of the result unless the construction of the result is feasible, nor profitable to test if the result has the desired benefits (P_T), unless it is semantically feasible.

We can regard sequence I as an elementary model of systemeering. The only task lacking from it is implementation, which could be attached at the end of the sequence. The model is also inadequate for another reason, namely that all the design tasks would be performed before any test.

The third construct is concerned with the three main aspects of the continuous control which is imposed upon all data processing in a purposeful system (see publ. 4). This means that we make decisions concerning the aspects P, S and C in this order and at the same time iterate the sequence I:

decision of aspect P ↓ decision of aspect S ↓ decision of aspect C ↓

(II) $P_D S_D C_D C_T S_T P_T \quad P_D S_D C_D C_T S_T P_T \quad P_D S_D C_D C_T S_T P_T$

The main activities in sequence II are P_D from the aspect P, S_D from the aspect S and C_D from the aspect C. The activities S_D and C_D from the aspect P will be performed preliminary in order to determine whether it is potentially possible to build up a data system fulfilling the pragmatic objectives and requirements. The decision to interrupt systemeering can then be made after the tasks of the first subsequence of II.

Considering the transfer from first subsequence of II to the second, we can conclude that it has been found at the end of the first subsequence that the design of the new data system fulfill the pragmatic requirements (P_T). After that we should again carry out pragmatic design (P_D), now within the aspect S. For the reasons stated above, we shall ignore that task here, and by the same arguments we shall also ignore P_D and S_D from the aspect C.

By applying to the subsequences of the P, S and C decisions in connection with sequence II the term P, S and C main phases, and by denoting them with upper indices, we obtain sequence III:

(III) $P_D^P S_D^P C_D^P C_T^P S_T^P P_T^P \quad S_D^S C_D^S C_T^S S_T^S P_T^S \quad C_D^C C_T^C S_T^C P_T^C$

This sequence is characterized by stress on the production of decisions and information. Each main phase produces suitable information for some decision maker on the data system under development. Simultaneously, iteration at the highest level is realized in this construct. It must be seen that this model is totally independent of organizations, and that a person or organizatiory unit can be engaged in more than one activity depending on the relation with the data system under development.

We found above that the implementation was not accounted for in our model of systemeering. After the addition of this (the fourth construct[1]) and the corresponding testing tasks C_T, S_T and P_T to sequence III, we obtain our final sequence with the implementation main phase included:

(IV) $P_D^P S_D^P C_D^P C_T^P S_T^P P_T^P \quad S_D^S C_D^S C_T^S S_T^S P_T^S \quad C_D^C C_T^C S_T^C P_T^C \quad BC_T^B S_T^B P_T^B$

The tasks of IV concern the whole data system, and it is this that is called the PSC systemeering model. We now need additional information about tackling of subsystems.

Subsystems of a data system

The partition into subsystems is crudely presented in Fig. 2.

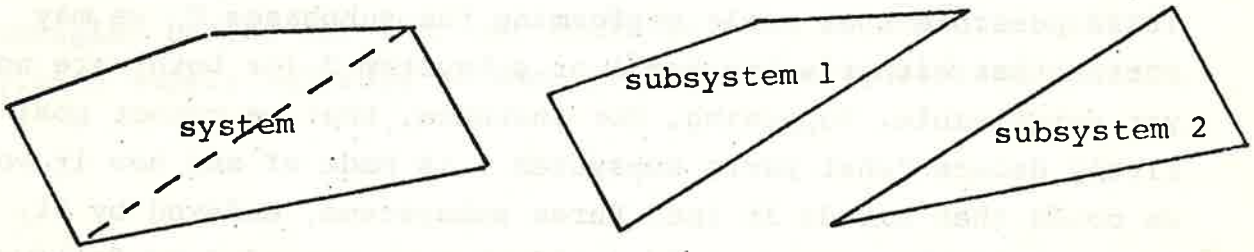


Fig. 2. The partition into subsystems

The partition process demands that the subsystems together should have the same external properties as the original system. A system is often divided into subsystems because the system itself is imperceivable, ie. too large to comprehend, and that the "smaller" subsystems can therefore be mastered more efficiently. Generally the subsystems are connected with each other in ways that are not included in the external properties of the original system. The "external properties" of a system refer to the relationships between the system and its environment.

---

1) see the same basis as in the second construct

Let us suppose now that in the systemeering task C_D^P we perceive it profitable to divide the data system into two subsystems. The hierarchical PSC structure, sequence I, is then applied to both subsystems. Sequence IV is then extended as follows:

$$P_D^P S_D^P C_D^P C_T^P S_T^P P_T^P \quad S_D^S \ldots$$

of subsystem 1 $P_D S_D C_D C_T S_T P_T$

of subsystem 2 $P_D S_D C_D C_T S_T P_T$

Task P_D of subsystem 1 represents its pragmatic design. Subsystem 2 is then allotted to the environment of subsystem 1 and can exercise certain pragmatic demands upon it. The other possible tasks also have to be interpreted in a corresponding way; ie subsystem 1 is now regarded as a system subject to design and testing. In due course the same also holds for subsystem 2.

It is possible that while performing the subphases C_D we may notice that either subsystem 1 or subsystem 2 (or both) are not yet perceivable. Supposing, for instance, that we cannot positively deduce "what parts subsystem 1 is made of and how it works", we could then divide it into three subsystems, endexed by 11, 12, 13 respectively. Sequence IV would then be extended as follows:

$$P_D^P S_D^P C_D^P C_T^P S_T^P P_T^P \quad S_D^S \ldots$$

of subsystem 1 $P_D S_D C_D C_T S_T P_T$

of subsystem 2 $P_D S_D C_D C_T S_T P_T$

of subsystem 11 $P_D S_D C_D C_T S_T P_T$

of subsystem 12 $P_D S_D C_D C_T S_T P_T$

of subsystem 13 $P_D S_D C_D C_T S_T P_T$

Our implication is that subsystems have to be divided sufficiently deeply in the C_D tasks, and that the sequence I ($P_DS_DC_DC_TS_TP_T$) has to be applied simultaneously and repeatedly.

The division into subsystems depends greatly on the person who carries it out. The more relationships there are between the subsystems, the more clarification and description work the operator has. Therefore the partition into subsystems should be carried out in such a way that each subsystem has as few connections with its environment as possible.

The other conclusion to be drawn from this discussion is that it is not possible to give a fixed phase-division model, applicable to every case of systemeering. The model presented here is flexible and should be applied according to situation. The latter point argues in favour of the view that there is no reason for learning the task list of sequence IV by heart, for instance, but one should rather try to adopt the constructional principles behind it. Then one would be able to construct phase division models suitable to each particular systemeering project.

III ON THE HIERARCHICAL CONCEPT STRUCTURE OF SYSTEMS DEVELOPMENT (OMITTED)

Figures 3-5 and appendix 4 contain outlines of the hierarchical concept structure of data systems development (DSD) and its management. The structure is DSD centred and was drawn up for further analysis and synthesis in the seminar. Appendix 5 contains the vocabulary of section 20 of the international standardization work (information system characteristics and development). Appendix 6 gives a set of models for decision criteria in systems development. Some of these are in Finnish and will be translated during the seminar.

ON THE CONCEPTUAL FOUNDATIONS OF SYSTEMEERING -A SEMIOTIC APPROACH

Mikko Klemola and Kalle Lyytinen
University of Jyväskylä
Department of Computer Science

ABSTRACT

It is widely agreed that systemeering lacks a well-defined set of basic concepts. In this paper an attempt is made to develope some theoretical concepts of systemeering using semiotics as the frame of reference. The starting point is the notion, that the so-called information requirements of the end user are based on the knowledge he has attained about the objective reality as expressed in his specific professional language and concept structures (knowledge) he needs in his activities(practice). The nature of systemeering is described as an analysis of semiotic relations of the specific professional language and as transforming this professional language into the object language of data production. Models needed in and conditions of the process transforming are defined. Some basic concepts needed in the development of coherent systemeering theory are given.

KEYWORDS Systemeering, System design, systems work, data, data system, data processing system, data models, data base, semiotics, data processing science

1. INTRODUCTION

The formulation and development of the basic concepts of systemeering and their relationships, i.e. the system of concepts of systemeering presuppose a profound insight into the original nature of systemeering. In the following we shall examine the basic concepts of systemeering departing from the presupposition that systemeering is a process of transforming a professional language into an object language of data production. In the process of transforming the semiotic relations of professional language - pragmatics, semantics, sigmatics and syntactics - will be defined and the physical basis and organization of object language will be designed.

So far the definitions of data processing and data processing theory have not succeeded to give the deeper understanding of the basic nature of data processing. The definitions do not give insight in to what the data processing is in the light of scientific research and theory construction. Our starting point is that data processing can be interpreted as operations on object language and these operations require human labour. Therefore we propose (1) that the term data production should be used about systematical labour(process) which is based on the syntactical properties of object language and (2) that the term data production system be used about physical system generating physical processes directed to the signs of object language. These resources and activities are often denoted by the term data system or data processing system. In this paper we will give the term data system a semiotic interpretation, i.e. we will define data system as a linquistic unit.

The existence of an object language - the object of data production - as a symbol system requires an analysis of the nature and the tasks of this language in a social system. According to our view the analysis is based on semiotics, logic and epistemology, i.e. concepts of philosophy. As far as we can see, the slow development of data processing theory is due to the lack reliance on the traditions of philosophy in the research of the relationships of language, objective reality, human knowledge and thinking. The situation is this irrespective of the fact that data processing science (informatics) studies data, language expressions, and its processing.

Firstly, the philosophical framework of our research is to be found in the following studies : Wittgenstein,1922;Klaus,1965,1972a,1972b,1972c,1974;Schaff, 1962;Kotarbinski,1961;Nauta,1972:Carnap,1956;Eco,1976:Klaus&Buhr,1974. The following researchers in data processing science have also developed semiotics, and in this sense can be considered as source for our thinking: Stamper,1973,1975; Zemanek,1974,1975;Slamecka,1968;Kent,1976;Kämäräinen,1977.

Secondly, field of research influencing on our theory is the study of defining data models and language structures (schemes) needed in the modelling of reality and professional languages: Senko,1976,1977;Codd,1971,1974;Biller& Neuhold ,1977;ANSI/X3/SPARC,1975;Sundgren,1973;Todd et al,1976;Mylopoulos& Wong,1976;Kerschberg et al,1977;Peltola et al,1978.

Recently, also artificial intelligence is studying data models.
It attempts by developing methods - language structures - for the representation of human knowledge in computer systems: Mylopoulos&Roussopoulos,1975;Weber,1976:, Mylopoulos&Wong,1976.

Another field of research investigating problems of representing knowledge by data is the study of abstract data types and their definition in programming languages:Minsky,1974,1976;Smith&Smith,1977.

The third source of our research is the application of system models and systems thinking for the design of data production systems: Kerola&Järvinen,1975; Lundeberg&Andersen,1972;Sundgren,1973;Langefors,1971.

The purpose of our research is not to develop - at this stage - new methods or tools for systemeering, but we suggest - that the increase of theoretical knowledge of systemeering and its nature is - in the long run - the best way to develop appropriate systemeering methods and tools.

This paper is divided into number of chapters in the following way:

In the second chapter we shall briefly define the semiotic concepts needed in our study and various kinds of languages.

In the third chapter we shall deal with the different counterparts relating to topics in semiotic research: objective reality, mind and language.

In the fourth chapter within the framework developed in the previous chapters an attempt is made to define the basic concepts required in constructing a systemeering theory and to characterize the objectives of systemeering and the tasks and work phases involved in it.

In the fifth chapter, some conclusions concerning the nature of systemeering and its research are drawn and suggestions for furter study are made. At the same time we try to determine the location of data processing science in the system of sciences and the research problems characteristic to it.

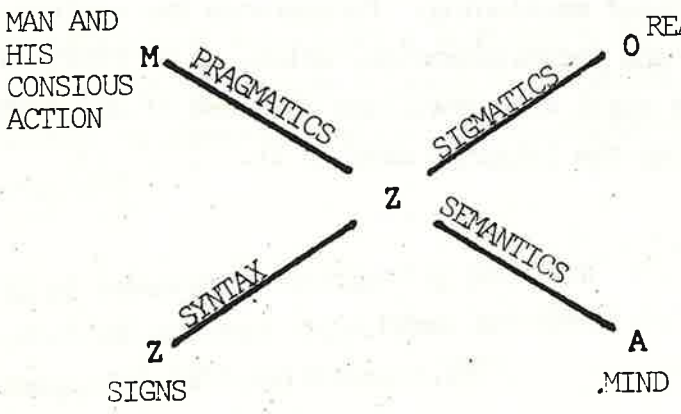
This paper is based on a larger research report in preparation. In that report we discuss the characteristics of data production as a semiotic system processing and producing sentences in the object language derived from the professional language. Therefore in this paper certain essential theoretical problems will be put aside or given just a brief mentioning. Furthermore the content of this paper can contain contradictions and inconsistencies, which, as we hope, will not come up in the final report. We shall with gratitude make use of all comments on the contents of this paper or the language used in it.

2. BASIC CONCEPTS OF SEMIOTICS

According to materialistic thinking,(objective) reality exists independently of human beings and their consciousness. The two expressions of consciousness interpreted as a reflection of objective reality, are human thinking and language. Materialistic semiotics is a branch of science studying a general theory of signs and symbol systems,i.e.from a materialistic point of view a theory of languages.

2.1. Subjects of semiotics

One way to analyze the subjects of semiotics is illustrated in figure 2.1.



O Objective reality

A Thinking and knowledge as a reflection of objective reality

Z Language(s) corresponing to thinking and knowledge and its signs

M Cognizing and thinking human being using language

Fig.2.1. Components of the Semiotics and their relations

In the following we shall examine the significance of various topics of research in semiotics as illustrated in Fig. 2.1.

(1) Signs of the language in relation to other signs of the language (the syntax of language) relation R(Z,Z)

The syntax of language defines the set of rules which determine whether a specific sequence of signs is to be considered a well-formed formula(expression) of that language, furthermore, it defines the conditions for forming new expressions from the existing ones. These rules and conditions are enumerated in the grammar of language[1)].

(2) The language and its signs in relation to mind (semantics of language) relations R(Z,A),R(A,Z)

The relationship between signs of language and the mind is bidirectional and reciprocal. The first relationship R(Z,A) means that the language objectities thoughts i.e. to express a thought you always need the language. Language is "an immediate actuality of thought". The second relationship R(A,Z) implies that the sense (meaning, connotation) of a language expression is a thought, a mental image reflected by this expression.

(3) The language and its signs in relation to the objective reality (sigmatics of language) relations R(Z,0),R(0,Z)

In terms of sigmatics in the relationship R(0,Z) language expressions are said to refer to (denote) the objective reality. The relationship R(Z,0) suggests that the signs of the language stand for the objects (things, states of affairs) in reality. Unless semantics were in a way included in it sigmatics would remain as an uncomprehensible abstraction. Language expressions share two characteristics. On the one hand they are capable of referring to reality only with their sense. This is their subjective feature. On the other hand, they also have their objective part i.e. they refer to reality. They are not only names of thoughts or mental images.

(4) The language and its signs in relation to man (pragmatics of language), relations R(Z,M),R(M,Z)

In terms of pragmatics the relationship R(Z,M) implies that man can formulate or construct language expressions. The opposite relationship R(M,Z) means that a human being can utilize language expressions created by himself or by other human beings.

1)Here we make distinction between a certain syntax of a certain language and a theory of these syntax structures i.e. syntactics. The same distinction is not made in semantics,sigmatics or pragmatics.

2.2. Definition of language, language systems

In this section we shall examine in a more detail the concept of language and its specifications: natural language, professional language, formal language and matelanguage.

We shall define language (or natural language) in general as follows:

Def.2.1. Language is a continuously evolving system of verbal signs established by the needs of social life, especially of production and used by human beings in the process of cognizing objective reality in order to formulate their ideas and to communicate them as well as their emotional, volitonal and aesthetical experiences and to preserve the knowledge thus achieved. (Klaus, Buhr, 1974; cf. also Schaff, 1962)

At this point, we would like to emphasize that natural language is universal. This feature of natural language gives us possibility to talk about all human experience and knowledge.

The continuous and deeper specification of science and production presupposes that natural language becomes more sophisticated: professional languages are created which differentiate more or less from natural language or from other professional languages. In professional languages, however, we always need language expressions originated from natural language and other professional languages. It is only their sense that can or have to be changed.

Professional language will be defined as follows:

Def 2.2. Professional language is a language whose expressions are meant for expressing things and states of affairs carried by them for specific e.g. professional purposes.

Such professional languages would include, among others, the languages of economists, accountants, jurists, ADP-specialists. When communicating with each other people in various occupations must "translate" expressions formulated in their own professional language into ordinary natural language expressions conveying the same information. This is /made/ possible by the universality of natural language. Natural language can be interpreted as a metalanguage of all professional languages.

A very special and important case of professional languages is the class of formal languages, which can be defined as follows:

Def.2.3. Formal language is a language , where all signs(symbols),form rules, definition rules,reproduction or reformulation rules i.e. inference rules and denotation(signification) rules are explicitly defined.

Various languages used in science are formal languages. On the contrary, professional languages used in production very often do not have typical characteristics of formal languages. Therefore, one of the main problems in systemeering is to formalize a particular professional language so that it can be processed by computer. In order to formalize a language a metalanguage is needed.

Def. 2.4. Metalanguage is a language by means of which we can talk about a professional (object) language. We can

(a) express what can be said in the object language
(b) enumerate and name signs,expressions etc. of the object language
(c) talk about expressions of the object language the properties of expressions and their relationships
(d) formulate form rules,definiton rules,reproduction and denotation rules about the object language.(Klaus&Buhr,1974)

A specific professional language i.e. an object language and its metalanguage are manifested on semantic levels. Semantic levels will be defined as follows:

Def.2.5. Semantic levels are

(a) The 0th level consists of things,properties of things and relations between things. This level is often also denoted in terms the object system of language or, in Predicate Logic the nonempty domain D of individulas of the interpretation model.
(b) The first level consists of signs (symbols) of language and sentences describing things and reflecting states of affairs on the zero level. Languages on this level are called object languages.
(c) The second level consists metalanguages by means of which we can talk about languages on the first level i.e. /about/ object languages.
(d) The third level consists of languages by means of which we can talk about languages on the second level i.e. /about/ metalanguages of the object languages.

The relations between various kinds of languages are illustrated in figure 2.2. in the form of a summary.

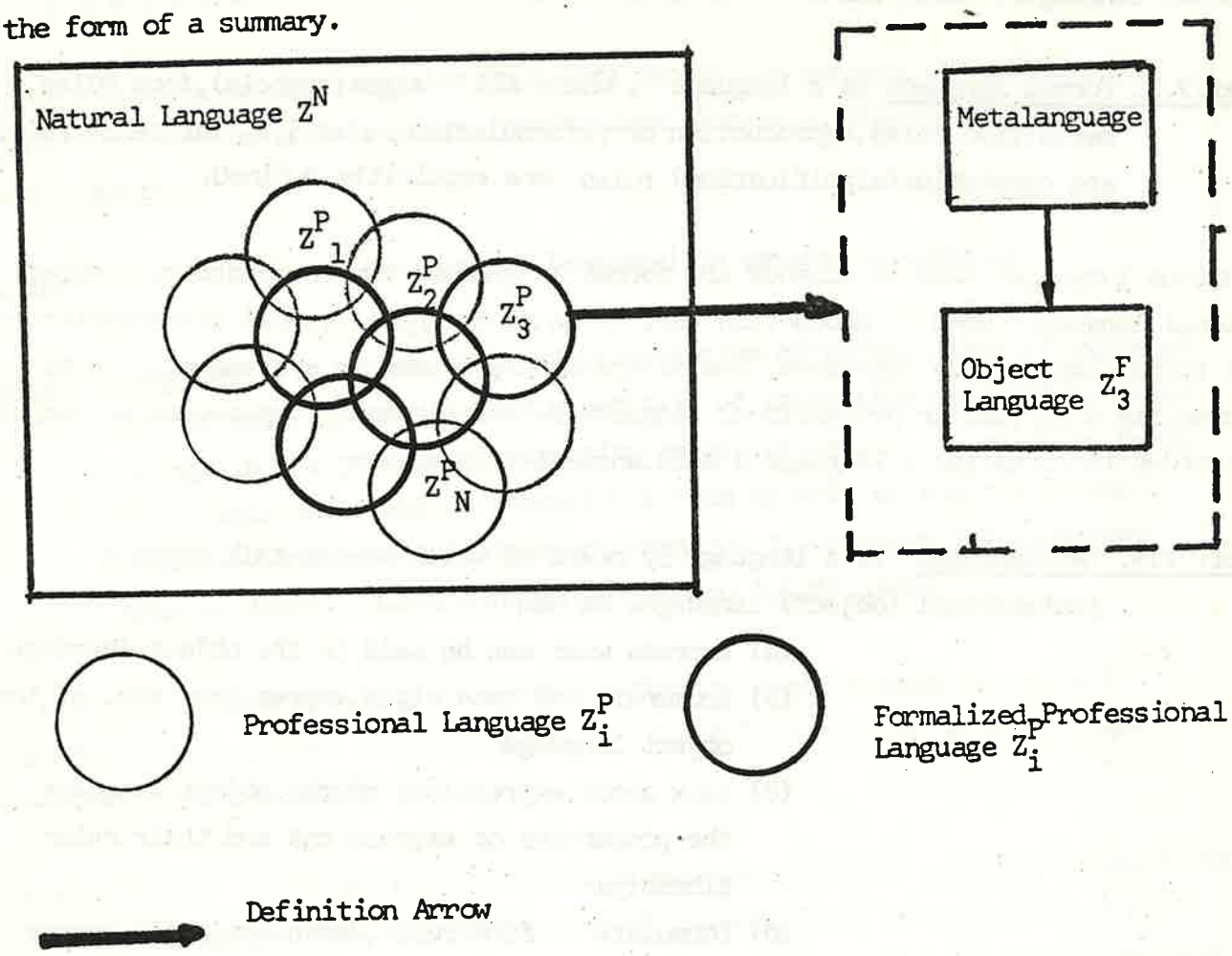


Figure 2.2. Relationships between languages

One will note from figure 2.2. that professional languages do not make a partition of natural language but are containing in a very complex way in each other. They do not compose a hierarchical structure but a much more comlicated "fuzzy" system structure. The language of mathematics, for example, is based on the language of logic; the language of physics derives mainly from the language of mathematics etc. This dependency illustrates different levels of abstraction used in formulating these languages and thus naturally affects the sphere of application of these languages.

3. OBJECTIVE REALITY, THINKING AND LANGUAGE

In this chapter we shall examine briefly the structure of the triple of the objective reality, the mind and the language.

3.1. Objective reality

We experiance objective reality through states of affairs (situations) of objective reality, which exist independently of us.

Def.3.1. A state of affairs (situation, fact) means that a relation exists between things, that a thing or a relation owns a property, that a property of a property is owned by a thing or a relation, that a property is owned by a class of things or relations etc.

Def.3.2. A thing is a system of (relatively) invariant properties belonging to objective reality. It exist independently of human mind. Through its properties a thing has relations to other things.[1)]

A thing must not necessarily be a physical or a material entity. It can also be an abstract entity (e.g. royalty, account, patent etc.)

3.2. The structure of krnowledge in terms of human mind

Some states of affairs are reflected to a human mind. These reflections will be called singular propositions.

Def.3.3. A singular proposition in the mind (a singular statement, a singular assertion) is a reflection of a state of affairs.

1) In defs. 3.1. and 3.2. we can find a circular definition. This is however a dialectical contradiction of our cognition. A thing without properties is a no-thing - antithesis of a thing. (cf. Hegel, 1961)

A singular proposition asserts always that a certain state of affairs is existing.

Singular propositions do not exist without general (universal) concepts.

Def.3.4. A concept is a mental generalization of a thing,a class of things, a property or a relation.

Singular propositions and concepts can always be interpreted as concept structures.

Def.3.5. A concept structure is a structural association of concepts or propositions. It reflects the foundations of forming a concept,i.e. definition,predication of properties and relations to a class of things, laws between the state of affairs,singular existing states of affairs and operating with the concept structures(forma of thinking).

Information is another term for a concept structure.

3.3. The structure ofknowledge on the level of language-sentences and terms

Def. 3.6. Data is a general term denoting to all language expressions as signs,words,sentences,formulas,figures,charts etc.

The linquistic counterpart of a concept is always a term i.e. a word or a group of words. The linquistic counterpart of concept structure i.e. a definition of a concept or a singular proposition etc. is always a sentence. The linquistic entity defined by the words and their order in a sentence analyzes the corresponding concept structure.

Def. 3.7. An elementary sentence is a sentence which does not contain other sentences.

Def.3.8. A language expression needed to describe a state of affairs is always a sentence, which depicts the semantic-sigmatic core structure of the corresponding singular proposition. A sentence has a semantic-sigmatic core structure if one can not remove anything from it without destoying its ability to describe the state of affairs,i.e. the model of the

sentence. A sentence describing a state of affairs is not necessarily an elementary **sentence .**

Def.3.7. defines the sentence as a syntactic unit (cf.elementary message) and def. 3.8. depicts the semantic sigmatic interpretation of a sentence.

4. ON THE CONCEPTS OF SYSTEMEERING

Any scientific theory serving practice has to explain its own existence, not only in terms of the conceptual starting points but in terms of the application of the theory. We are not very convinced that the latter task has often been succesful apart from the former in the systemeering research. For that reason, we shall have to test the concepts of systemeering theory by abstracting to its conceptual basis not that much in terms of design -theory, but rather as related to human practice.

To start from , we have the fact that the final outcome of systemeering is part of human practice, part of production. That part of production which the systemeering theory in the first place is being applied to, will be called data production.

Def.4.1. Data production refers to the functioning of the socio-technical systems. This is directed to the processing of the material objects as signs so that signs and combinations of signs so produced can be interpreted as well-formed sentences (their syntax,semantics,sigmatics being accurately defined or known) of a certain object language.

With the term"data production ", we are referring to the systematized nature of the process in question - to its form that is consciously ordered. On the other hand the definition emphasizes lack of semantic and sigmatic relationships of data in its production i.e. when the data is the object of labour. The term data has only an operative meaning in the data production.

Def.4.2. The term data has an operative meaning in a certain context provided that syntactic rules needed for its processing are known.

However, the production of data is not sensible if the data has not ideational meaning,too .

Def.4.3. The data will have an ideational meaning in a certain context if the sense of the data or/and the objects of reality referred to by the data are known.

In this way the data production is a manifestation of the division of labour - the production (manifestation of the operative meaning) and the interpretation and the use of data (manifestation of ideational meaning) taking place at the different parts of labour community.

The process of organizing the data production (data production is often called data processing funtion cf. Kerola&Järvinen,1975) is generally called systems work. The most important subphase of systems work is the systemeering.

Def.4.4. Systemeering refers to the definition of that part of certain professional language that can be formalized and to its transformation into the object language of data production.

Our definition emphasizes the connections of systemeering with practice. The starting point of systemeering is a concrete language and its outcome is a concrete language. The forces that trigger systemeering process are the iron laws of economics.

The presupposition for the process of organizing of data production is also the explicit definition of the operative meaning of language or the definition of a certain part of a professional language as a formal language. The part of professional language as signified by the definition is called data system.

Def.4.5. A data system is the wholeness of actual and potential sentences of certain formalized professional language (object language) which expresses or can express a certain part of objective reality a data system desciption space or an object system.

It must be noted that the definition emphasizes
(1) the character of data system as a sign system; therefore we can study pragmatics, sigmatics, semantics and syntax of a data system.
(2) the continuity of data system - through potential class of sentences in object language it has certain invariant properties.

The actual existing sentences at a certain point of time of data system could be called the state of data system.

The object system, which is expressed in the professional language and the systemeering object system serving as a concrete starting point of systemeering, are, according to our opinion, completely different concepts. This is due to the following reasons.

(1) The object system explains only the sigmatics of the professional language. In systemeering the professional language and parts of it have to be examined as a certain wholeness having a certain kind of pragmatics, semantics, sigmatics and syntax.

(2) Professional language is not isomorphic with its object system. For that reason, an object system cannot explain all sides of professional language and vice versa.

(3) The task of systemeering is not at all to form a new professional language but to transform the existing ones into data production and to give it in this way a new form.

The professional language and the corresponding concept structure have differentiated in their functioning. This differentation is an outcome of a historical process. It cannot be reached in the systemeering ad hoc.

For these reasons we give the following definition:

Def.4.5. A systemeering object system is a social system or constitutes a part of it. When aiming at certain objectives it is in interaction with a certain part of objective reality (i.e. object system); to cognitize and control this it needs a certain concept structure (world view) and a professional language(s).

The definition emphasizes that professional language and the corresponding concept structure build up a whole and both are the prime objects and necessary presuppositions of systemeering. It also points out that the professional language is concrete - it does not exist without the users of the language, without a social community and its members.

The user is defined in the following way:

Def.4.7. The user is a member of a social community, who in his practice makes direct use of a certain professional language.

The systemeering process itself has to be examined in the first place as the production of metalinquistic expressions describing the data system being defined at different levels of detail. This production has to take into consideration different possibilities of realization of data production. It chooses tha data production system being capable to data production and produces the metalinquistic expressions of data system taking into consideration this realization model.

The expressions produced about the data system can be divided logically into three models. We call these models Logical Data System Model (LDSM), Realization Model (RM), and Physical Data System Model (PDSM). LDSM and PDSM are metalinquistic expressions of the professional language. RM is a linquistic expression (on the first semantic level) of the things regulating the definition of PDSM.

Def.4.8. LDSM is a metalinquistic expression of the formalized class of potential sentences of a certain professional language i.e.of data system and of syntactic,semantic,sigmatic and pragmatic properties of these sentences.

Follwing properties charecterize the modelling of LDSM:

(1) The professional language is examined as the logical system, consisting of sentences in the semiotic frame of reference.
(2) The production of LDSM starts from the concrete systemeering object system.

(3) In the metalanguage, the sentences of the object language are named at the level of sentence equivalence classes so that the properties and the relations of sentences in the object language can be exhaustively described. For the naming and describing of the sentences the sentence variable (sentennial function, propositional function) (Klaus,1972b) will be used.

(4) In the metalanguage e.g. the following will be described: the definition rules of terms in the sentences,the form,reproduction and denotation rules of the terms or the sentences and the inference rules (deduction rules) which are needed for producing new sentences in the object language.

(5) LDSM does not describe the physical form of sentences in object language.

The realization of LDSM presupposes further the selection and description of material dataprocessors and data carriers. This description or model is called RM.

Def.4.9. RM descibes the data processors and data carriers used in data production and the material properties and material relations of these data processors and data carriers and their functional characteristics.

Data processor is defined as follows:

Def.4.10. A data processor is a material thing being able to process data by obeying insrtuctions. (cf.Kerola&Järvinen,1975)

It is characteristical for RM that:

(1) different realization possibilities often have very different economical and practical effects.

(2) Economic resources put severe restrictions on the realization possibilities.

(3) In general, there are many alternative dataprocessor/data carrier combinations which can realize the LDSM.

(4) RM often exists even before systemeering starts i.a. system consisting of data processors and data carriers i.e. the data production system which will perform functions in data production has been given.
(5) After the data production system has been defined, other parts of RM consists of the functional descriptions of the data production system.

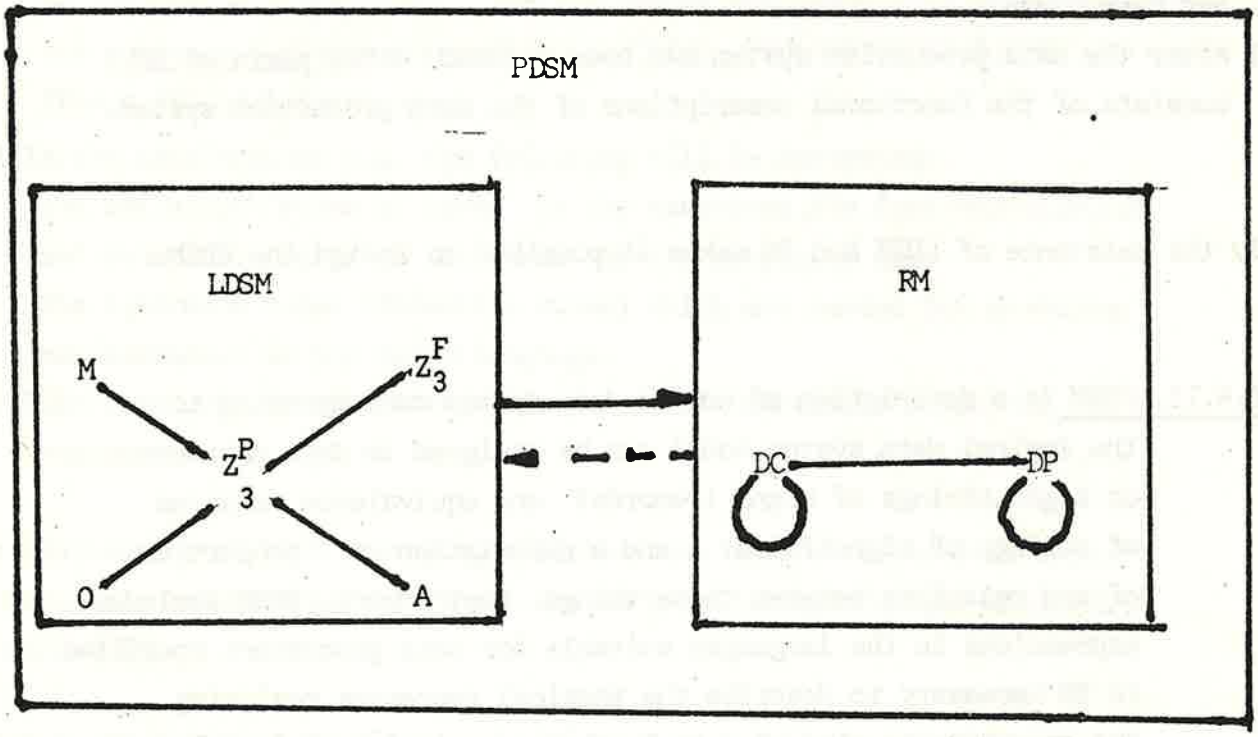
Only the existence of LDSM and RM makes it possible to design the PDSM.

Def.4.11. PDSM is a description of how a data system corresponding to the logical data system model can be assigned to data carriers as signs, strings of signs (records) and equivalence classes of strings of signs (files), and a description of properties of and relations between these things. Furthermore, PDSM includes expressions in the languages suitable for data processors specified in RM necessary to describe the physical processes realizing the operative meaning of data in this physical symbol system.

The design of PDSM is characterized by the following traits:

(1) The description languages (metalanguages) used in the model are dominantly determined by the RM.
(2) There exist many alternative ways to assign the LDSM to data carriers and data processors dictated by RM i.e. to design the PDSM.
(3) Because only this model "materializes" data, it includes many descriptions which are not logically deducible from the LDSM (e.g. gathering of data, transferring of data from a data carrier to another, transferring and maintaining of data carriers, conversions etc.)
(4) The necessary condition for designing the PDSM is that LDSM must be "compiled" to PDSM.

The crude frame structure for formulating the various data system models can be illustrated by figure 4.1.



DC = Data Carriers

DP= Data Processors

Z^P_3 = Professional language being the object of systemeering

Z^F_3 = Formalized version of professional language (object language)

⟶ = Mapping arrow

Fig.4.1. The objects of description in various data system models

The LDSM describes the professional language as a system composed of sentences. The RM depicts the data carriers and data processors used to physically implement the data system. The PDSM is a description of physical representation and processing of sentences in object language consistent with the LDSM using RM.

The schematization of models serves not only for analysis of systemeering process but clarification of the nature of data production ,too. Data production is determined by social practice and economical resources of organization(social system);the social world view (Weltansschauung) i.e. a frame of reference and propositions formed within it; the professional language used to represent propsitions and knowledge in general; tools and methods for organizing data production (e.g. computers,programming languages,operating systems,DBMS:s,varios kinds of metalanguages for describing data system).

Systemeering creates also so-called "side-effects" i.e. effects on the social system which are not analyzable on the semiotic level. These side-effects include:

(1) The net-benefits achieved by designing a new data system (economical level);
(2) The consumption of resources;
(3) The effects on the "social life" of systemeering object system (organizational changes, unemployment, resistance to changes, job satisfaction, changes in jobs, learning by doing etc.);
(4) The effects on physical environment of social system (new machines, rooms, ergonomical effects etc.)
(5) Other effects (juridical, aesthetical etc.)

These side -effects are very often important, we can say decisive, for a succesful finishing of systemeering project.

5. CONCLUSIONS

In this paper we proposed a semiotic approach for developing a coherent systemeering theory. The main problems of systemeering in terms of semiotics include

(1) analysis of semiotic relations of the professional language, the formulation of professional language as a formal language;
(2) design of the physical basis and organization of a formalized professional language i.e. object language (data carriers) - selection of data processors - performers of physical operations on sentences of the object language, design of these operations;

The next subphase in systems work is to implement the PDSM specified in (2).

We record the following facts as benefits from using semiotics and the theory of formal languages in our approach. It makes possible

(1) to present for the concept data system a semiotic interpretation as a whole consisting of sentences of the object language;
(2) to define descriptions of the data system i.e. system documentation as desciptions in metalanguage about object language;
(3) to point out in detail what is the primary task of data production. This is the production of sentences in object language conveying singular propositions. This transfers the main emphasis from the variable oriented (or record oriented) i.e. from the data-item oriented study dominating in the computer science to sentence oriented study necessary for theoretical research about data production systems (data systems).

In this paper we have only presented some general and necessary basic concepts for theory construction. In future we shall examine more accurately and thorougly the following promising areas of research:

(1) side-effects on the social system of the transformation of the professional language into the object language of data production; a coupling of an analysis of the side-effects to the phases of a transformation process.

(2) descriptions in metalanguage, various kinds of metalanguages and models designed with them:
  (2.1) the so-called transfer-models inside and between the LDSM and PDSM, the number and the nature of them. On the basis of preliminary surveys, the means by which one can define the syntax of object language and the nature of transfer from a syntax (manual, automatical compiling) to another is very decisive.
  (2.2) taxonomy of data system models.
(3) the nature and structure of object language, especially its semantics and sigmatics.
(4) the logical nature of questions in object language. Metalanguage descriptions of queries formulated in object language. Analysis semantics and sigmatics of questions.

The frame of reference outlined in this report makes it possible to mention some aspects of the nature of data processing science (informatics) and research in systemeering. Data processing science can be defined as a science studying and developing theoretical constructs about metalanguages of object languages i.e. a theory of properties, structure and application of metalanguages. On the other hand data processing science has characteristics , which include in data processing science empirical objects as subjects of research and empirical constructs of them presented in some theoretical language viz. study of physical realization and processing of sentences in object language defined by metalanguage expressions.

We think that the starting point of our research can give rise to many original contributions. To our knowledge, this is one of the first attempts to apply consistently the concepts of semiotics, logic and epistemology to the research of data systems and to the definition of systemeering. (Others cf. Kämäräinen, 1977; Kangassalo, 1974) In spite of the preliminary nature of our approach, many fruitful approaches to develop systemeering concepts have been proposed.

ACKNOWLEDGEMENTS

The authors would like to acknowledge stimulating discussions on the relevant topics of this paper with Professor Eero Peltola (University of Jyväskylä) and Associative Professor Pertti Järvinen (University of Tampere). We would also like to thank Mrs. Marjatta Lehtinen for her helpful corrections of the English of this paper.

REFERENCES

A. PHILOSPHY

Wessel (1976) : Logik und Philosophie, VEB Deutscher Verlag der Wissenschaften,Berlin

Wittgenstein (1922) : Tractatus Logico-Philosphicus (deutsch und engl), London

Klaus (1965) : Spezielle Erkenntnistheorie,VEB Deutscher Verlag der Wissenschaften,Berlin

Klaus (1972a) : Semiotik und Erkenntnistheorie,VEB Deutscher Verlag der Wissenschaften,Berlin

Klaus (1972b) : Moderne Logik,VEB Deutscher Verlag der Wissenschaften, Berlin

Klaus (1972c) : Die Macht des Wortes,VEB Deutscher Verlag der Wissenschaften, Berlin

Klaus (1974) : Rationalitet -Integration - Information,VEB Deutscher Verlag der Wissenschaften,Berlin

Schaff(1962) : An Introduction to Semantics,Pergamon Press,London& Warzawa

Kotarbinski (1961) : The scientific Approach to the Theory of Knowledge, Pergamon Press,Warzawa&London

Nauta (1972) : The Meaning of Information,Mouton,Hague&Paris

Carnap (1956) : Meaning and Necessity,The University of Chicago Press, Chicago&London

Eco (1976): A Theory of Semiotics,Indiana University Press,Bloomington,London

Klaus&Buhr (1974) : Philosophisches Wörterbuch,VEB Bibliogaphisches Institut, Leipzig

Hegel (1961) : Science of Logic,London

B. DATA PROCESSING SCIENCE

Stamper (1973) : Information in Business and Administration Systems, Batsford,London

Stamper (1975) : Information Science for Systems Ananysts, in Mumford& Sackman (ed): Human Choice and Computers,North Holland & Elsevier

Zemanek (1974) : What is Informatics,Management Informatics,1,2,43-52

Zemanek (1976) : The Human Being and the Automation, in Mumford&Sackman _____

Slamecka (1968) : The Science and Engineering of Information, in Finerman (ed) University Education in Computing Science,Academic Press, New York&London

Kent (1977) : New Criteria for the Conceptual Model, in Lockemann&Neuhold (eds): Systems for Large Data Bases,North Holland, Amsterdam, Oxford,New York

Kämäräinen (1977) : Tietosysteemi ja abstraktiotasoperiaate sen rakentamisessa ja kuvaamisessa; Tampereen Yliopisto,Matemaattisten tieteiden laitos,raportti A-8 ,(in finnish)

Senko (1976) : DIAM as a detailed example of the ANSI SPARC Architecture, in Nijssen (ed): Modelling in Data Base Management Systems, North Holland,Amsterdam

Senko (1977) : Conceptual Schemas,Abstract Data Structures,Enterprise Descriptions, in Morlet& Ribbens(eds): Proc. of Int. Conf. of Computing 1977,April 1977

Codd (1971) : Further Normalization of the Data Base relational Model, in Rustin (ed):Data Base Systems,Courant Computer Science Symposium,Prentice-Hall

Codd (1974) : Recent Investigations in relational Data Base Systems, Information Processing 74,North Holland

Biller & Neuhold (1977) : Semantics of Data Bases,the Semantics of Data Models,Information Systems,3,1 pp 11 - 35

ANSI/X3/SPARC (1975) : Study Group on Data Base Management Systems, Interim Report

Sundgren (1973) : An infological approach to Data Bases,Statistiska Centralbyrån,Skriftserie nummer 7,Stockholm

Todd et al (1976) : Relations and Entities , In Nijssen (ed) _____

Mylopoulos & Wong (1976) : Two Views of Data Semantics: A survey of Data Models in Artificial Intelligence and Data Base Management,A.I. Memo 77-2,Dec,1976, University of Toronto

Kerschberg et al (1977) : A Taxonomy of Data Models, In Lockemann&Neuhold (eds) ___

Peltola et al (1978) : Grammatical Data Base Model,Manuscript,Department of Computer Science,University of Helsinki,to be published

Mylopoulos & Roussopoulos (1975) : Using Semantic Networks for Data Base Management, Kerr (ed) : Proc. of VLDB,Framingham,Massachusetts,Sept 22-24,1975

Weber (1976) : " A semantic Model of Integrity Constraints on a Relational Data Base ,in Nijssen (ed) _____

Minsky (1974) : On Interaction with Data Bases, in Rustin (ed) ACM SIGMOD Workshop on Data Description,Access and Control

Minsky (1976) : Files with Semantics ,Proc. of ACM SIGMOD Conf. on Management of Data

Smith & Smith (1977) :"Data Base Abstractions,Data Aggregation ,CACM,June,1977

Kerola & Järvinen (1975) : Systemointi II ,Gaudeamus,Helsinki (in finnish)

Langefors (1971) : Theoretical Analysis of Information Systems,Studentlitteratur Lund

Lundeberg&Andersen (1974) : Systemering-Informationsanalys,Studentlitteratur, Lund,(in swedish)

Kangassalo H (1974) : Relaatiotietokanta, Savolainen V (ed.) DIFO-yearbook 3/1974, DIFO-tutkimus, Tietojenkäsittelyopin laitos, Tampereen yliopisto, Tampere, 1974 (in finnish)

Heikki Kämäräinen
University of Tampere

REPORT
1978-08-18

AN ANALYSIS OF DATA PROCESSING SYSTEM AND THE ACTIVITIES RELATED TO IT

0. INTRODUCTION

1. MAN IN HIS ENVIRONMENT - CONCEPTUAL TOOLS FOR THE ANALYSIS

2. DATA PROCESSING SYSTEM - AN AGGREGATE OF FOUR DIFFERENT KINDS OF SYSTEMS

3. THE ACTIVITIES RELATED TO DATA PROCESSING SYSTEM - AN ANALYSIS OF THE ENVIRONMENT OF DATA PROCESSING SYSTEM

FIGURES

To be published in Kerola,Klemola,Kämäräinen,Lyytinen (eds.), Summary report of systemeering research seminar in Tampere 21.-24.08.1978. Finnish Data Processing Association.

0. INTRODUCTION

The roots of this presentation lie in work with Data Dictionary/Directory Systems. The original objective of the work was to design and implement a theoretically sound Data Dictionary/Directory System. Data Dictionary/Directory System was conceived to be a system that handles documents on data processing systems (primarily documents on data) as well as supports the systemeering work and the data processing activities in various ways. The first goal was to construct the kernel of the system, i.e. the subsystem that handles the document data base.

The major difficulties were not felt to be of technical character - there were and there are many advanced DBMSs. The major difficulty lay in identifying, what to document and what for. Consequently the weight of the work shifted from the tools and techniques of documentation on one hand to the object of documentation and on the other hand to the activities utilizing the documents, see 16. Kämäräinen. At the same time, when this presentation tries to answer these difficult questions related to documentation, it also is an attempt to conceive, what the basic concepts in systemeering could be.

Chapter 1 analyzes man in his environment. Some of the basic categories that are involved in man's action are identified and discussed. This analysis forms the background and supplies the main conceptual tools for the two latter chapters. Chapter 2 presents a model of data processing system, according to which data processing system should be dissected into a system of models, a system of model descriptions, a system of data processing resources as well as a system of data processing activities. These basic interpretations of data processing system are discussed in greater detail. Chapter 3 gives an account on the

activities related to data processing system. This account is compatible with the model of data processing system and is based on the classification of man's interactions with his environment.

Acknowledgement I want to express my gratitude to Hannu Kangassalo, who has introduced the field of semiotics to me. I also like to acknowledge the comments of Pertti Järvinen on a draft of this paper as well as the numerous, rambling discussions with Pekka Tyllilä.

1. MAN IN HIS ENVIRONMENT - CONCEPTUAL TOOLS FOR THE ANALYSIS

In order to clarify and justify our analysis of data processing system and the activities related to it, let us first discuss man in his environment. The starting point for the discussion is the well known subject - object setting, which is expanded as the discussion proceeds, see fig. 1.1.

Subject is part of objective reality (real world). By objective reality we mean everything that exists in space-time continuum and that can (in principle) be reached by the senses of subject. Thus objective reality consists of the past, the present and the future. At every time instant real world is in one state (x) of all the imaginable states of the world (X). Every subset of X defines an imaginable state of affairs A of real world. By object we mean the part of real world, to which subject has turned his attention and which subject tries to observe or to change by his acts. /2. Ahmavaara, chap. 1,2/.

Each subject has consciousness. Consciousness is the realm of thought. Although the mental processes in consciousness could be proved to be subject to observation, in everyday

life consciousness can be regarded non-accessible to others but its owner. Real world is reflected in consciousness, i.e. subject is able to sense real world and acquire knowledge of it. Subject is, however, not only a passive observer of real world. Subject is also a conscious actor, i.e. based on his knowledge of real world subject is capable of changing real world; while acting on real world subject acquires new knowledge. Speaking (or any other comparable form of communicating) is one way, how subject can affect real world. /2. Ahmavaara, chap.1/.

Subject should not be treated in isolation from the cultural and linguistic community, the member of which he is. From this cultural and linguistic community he has inherited among other things a set of concepts, a native language and perhaps a few modes of using the native language e.g. mode of speaking and mode of writing. It is worth noting that concepts, language and modes of using language are the property of cultural and linguistic community and not the property of any individual subject. See fig. 1.2.; cf. 3.Alston; 8.Heath.

Cultural and linguistic community can be said to possess a set of concepts shared by virtually all members of the community. The concept of concept is ambiguous, and it remains useful precisely because of its ambiguity. Yet, it can be characterized by explaining, what we mean, when subject is said to have a concept 'x'. To have a concept 'x' is a) to know the meaning of the word "x" (to be able to use the word "x" and to be able to respond correctly when somebody uses the word "x") ; b) to be able to distinguish x's from non-x's or to be able to have images of x in consciousness ; c) to apprehend the properties which characterize x's and make them what they are (to be able to deal with x's). Having a concept is thus revealed in these capacities, the first one of which focuses on words (nominalism), the second one on thoughts (conceptualism) and the third one on things (realism). Nevertheless, con-

cepts should neither be identified with words, thoughts nor things.

These capacities are acquired by some process of comparison, selection and abstraction. By exposure to common environment, common education and social attrition concepts are transferred to the newcomers of cultural and linguistic community. The thus acquired concepts operate as a principle for the ordering of experience, the guidance of action, or the control of thought and speech. On the other hand the experience that subject gains affects his set of concepts. Sometimes the term concept is used as an abbreviation for the sense of words or phrases. Then the term concept is used in an abstract sense. When considering the meaning of words this practice suits us , too, and is thus adopted in this paper. /8. Heath/.

Language is also a property of cultural and linguistic community. Language can be characterized as a system of constituent symbols which provides an opportunity for constructing an indefinite number of aggregate symbols. Symbol is a sign that has meaning by convention; sign is something that stands to somebody for something in some respect or capacity. In a crude analysis the constituent symbols of language are words. The aggregate symbols are sentences; thus sentences are made up of words. Although words are the entities which have meaning by convention, only a sentence can perform a complete act of communication all by itself. The usage of language is determined by syntactical, semantical and pragmatical rules. In natural languages the rules are of both de facto and de jure character, whereas in artificial languages only of de jure character. /3. Alston; 5. Alston; 20. Nauta,48/. - (De facto rules state something, which happens, irrespective whether it ought to or not. De jure rules state something, which may or ought to happen, irrespective whether it does or not /20. Nauta,49/.)

It is important to distinguish between language and the employment of language. For each language there can be several modes of using language. Speech has always been the most readily available mode of using language. Others are mainly derivatives of speech, such as writing. The use of language takes place in space-time continuum and is (with the exception of thinking) observable. Language itself is not observable. It is an abstract system that guides the use of language. If the mode of using language is speech, then language can be regarded as an abstract system of sound types and the rules of their combination, which is exemplified in verbal activity and which is discoverable through an analysis of this activity. In the same way as language is distinguished from speech, it must also be distinguished from thoughts. Thinking can be viewed as one mode of using language. /3. Alston/. - From the above it is evident that language has a central, guiding role both in interpersonal communication and in thought. In addition it affects ordering of experience and the actions of subject.

A short discussion on the meaning of words and sentences clarifies the use of language. In the course of history a few basic types of theories concerning meaning have developed. Each of these is based on a real insight into the nature of language. The referential theories are right in arguing that language is used to talk about (refer to) real world - yet, not all words refer to real world entities in the same way as others. The ideational theories are right in arguing that language is used to express thoughts and ideas - yet, a word does not evoke a standard image in subject's consciousness each time it is used. The stimulus - response theories are right in arguing that words have meaning by virtue of the ways in which they figure in human activity (events function as stimulus for speech; speech evokes events) - yet, words do not figure in regular stimulus - response associations./ 4. Alston/.

The modern theories of meaning try to avoid the one-sidedness of the older theories. According to the modern theories the meaning of word is to be found in its contribution to a sentence, and the meaning of the sentence is to be found by observing, what the subject, who uttered the sentence, is doing, i.e. by trying to find out, what subject wants to achieve by uttering the sentence. These intentions of subject, and thus the functions of language (sentences, words) have been classified in various ways. According to one classification language is used in descriptive function (to describe states of affairs in real world; to give form to the utterance), in control function (to incite a specific behaviour; to indicate, what goals are worth pursuing; to prescribe performance) and in emotional function (to express experiences, feelings etc. of subject; to evoke a certain state of mind in somebody else; to establish and maintain a social contact). /4. Alston; 20. Nauta,50/.

So far our discussion has focused on objective reality and on cultural and linguistic community. Let us now turn the attention to subject. Subject's consciousness is the realm of thought. In most cases the mental thought processes can be considered as the use of language and the thought structures as linguistic structures (although the mode of representation is concealed). Thus the realm of thought and the realm of speech are comparable with each other in relation to the realm of language. Sometimes it is appropriate to postulate that the realm of thought is populated by mental images and that the mental processes are those of nonverbal imagery. (As a matter of fact, linguistic thought structures might be regarded as mental images of spoken or written expressions.) Still on other occasions it is well suited to regard the contents of consciousness as a set of dispositions. /24. Woozley/.

In subject's consciousness we can distinguish a set of concepts, language that he has adopted and view of the world,

see fig. 1.3. These are all partly inherited from cultural and linguistic community and partly formed on the basis of personal experience. The closer the concepts and the language of subject correspond to those of community the less difficulties subject has in his communication; the more truthful the view of the world is , the less difficulties subject has in coping with real world. The contents of consciousness as a whole guide the behaviour of subject.

View of the world[1] consists of a set of beliefs, a set of values and a set of norms. Beliefs are subjective probabilities concerning states of affairs, i.e. a belief expresses how probable a certain state of affairs is in subject's opinion. Values organize states of affairs in an order of preference, i.e. values express what subject finds valuable and worth achieving. Norms organize the acts of subject in an order of preference, i.e. norms indicate the course of action for subject. Norms are dependent on beliefs and values. /2. Ahmavaara,15 - 17,107 - 108/.

Fig. 1.4. shows an abstraction of subject. According to this abstraction subject can be dissected into four subsystems: receptors, emittors, memory and processor. Receptors are the organs by which subject senses his environment. It is via receptors that environment affects subject. There are different receptors for different types of influence streams that object is exposed to. Emittors are the organs by which subject affects his environment. There are different emittors for different types of influence streams that subject emits into his environment. Memory is the organ where knowledge of the past behaviour of subject can be stored. This knowledge can concern the stimuli, the inner activity and the responses of subject.

---

[1] View of the world was not discussed in connection with cultural and linguistic community as it seems to be more sensitive to the subcultures of community than concepts and language are.

If view of the world is regarded as a structure of thoughts, memory is the repository for view of the world. Processor is the organ for information processing. Processor receives its inputs from receptors, memory and from itself. The action of processor is guided by view of the world. Processor controls the action of receptors and emittors as well as the contents of memory. /14. Klir,157-158; 20. Nauta,78-,140-/.

Fig. 1.5. illustrates the ways in which subject interacts with real world. To support the classification of the interactions, real world is divided into two realms: material environment and symbol environment. In both cases the interactions take the form of either material or energy; in the case of symbol environment the interactions are interpreted as symbols. Subject interacts with material environment in three ways : by observing object (see fig.6), by acting on object (see fig. 6) and by being acted on by object. With symbol environment subject interacts by communication i.e. by exchanging symbolic messages (see fig. 6). /21. Nordenstreng,75-79/.

The discussion on man in his environment is summed up by repeating the basic categories which are involved in man's (linguistic) behaviour and by identifying some of the possible relationships between the basic categories. We have the categories of concepts, language, speech, thoughts and things, see fig. 1.7. The categories of concepts and language are abstract categories; they are both collective property of cultural and linguistic community. The categories of speech, thoughts and things are real world categories and thus an entity belonging to one of these categories is located somewhere in space-time continuum. - (Thoughts differ from spoken words and things in that thoughts are non-accessible at least to everyday observation.)

Fig. 1.8. illustrates the relationships between the basic categories. Four categories: things, concepts, language and speech are represented. The category of speech is interchangable with the category of thoughts. In the category of things we can distinguish particulars:objects, properties and relationships. Properties are always properties of objects. Relationships prevail between objects. In data processing we are not so much interested in objects of independent existence as in objects of description. Thus it is appropriate to postulate that all that exists is objects-that-have-properties and the property-less relationships between them. If a relationship interests us so much that we want to know some properties of it in addition to its existence, it becomes an object of description.

Corresponding to the structure of the category of things there are class concepts, quantitative concepts and relation concepts. All these concepts are general, in other words they are universals. Acquaintance with new concepts is mainly got (at least in the realm of systemeering) via concept definitions. If the concept in question has a real world referent, class concept refers to all the objects of the type and relation concept to all relationships of the type. Quantitative concepts have the structure of function. Quantitative concept relates the properties in question to a scale of measurement. In our illustration we have two class concepts CAR and MAN, one relation concept OWN and one quantitative concept LENGTH.

Words, which belong to the category of language, are related to concepts and things. Concepts are designated by words e.g. the concept of 'CAR' is designated by the word "CAR". If the corresponding concept refers to real world entities, the word refers to these entities ,too. Nevertheless, not all words designate concepts (i.e. give names to universals). In our illustration we have two types of words: proper names and values that do not desig-

nate concepts but are related to particulars. According to the no-sense theory,proper name does not designate any (individual) concept; it only refers to (individual) object. Object can be distinguished from other objects that belong to the same class by its proper name. In our illustration "JACK" and "THUNDERBIRD" are proper names. Values are results of evaluation. Property of object is evaluated by using a method of measurement . In measurement property is related to a point in the scale; this point is the value of measurement. In our illustration "5 m" is a value. - Finally, words are represented by the use of language e.g. by uttering appropriate sounds. It is worth noting that in semiotics often no clear distinction is made between abstract word and a space-time continuum bound representation of it.[1]

The account above can be subjected to ample criticism not the least due to the fact that it mainly treats words as names and considers the sole function of language to be description. However, the symbol strings that are manipulated in data processing and that are used in systemeering are in most case used precisely in the descriptive function of language i.e. to name objects and describe their properties and relationships (to state facts, to express knowledge). Therefore this account is applicable at least in some areas of systemeering.

---

[1] The above analysis of the relationships between the basic categories is based on /7. Bunge, chap. 2.2; 8. Heath; 12. Kangassalo; 17. Langefors; 23. Searle; 24. Woozley/.

2. DATA PROCESSING SYSTEM - AN AGGREGATE OF FOUR DIFFERENT KINDS OF SYSTEMS

When thinking of and discussing data processing system it should consciously and consistently be dissected into four different kinds of systems[1]: system of models, system of model descriptions, system of data processing resources and system of data processing activities, see fig. 2.1. Model defines data processing resources and data processing activity. Model description describes model; functional resources reach model via model description. When functional resources guided by model process data carrying resources data processing activity takes place. See 16. Kämäräinen, 2.1.[2]

This dissection is based on the basic categories of human (linguistic) behaviour (as analyzed in chap. 1 of this paper) and on the distinction between static and dynamic systems. Model belongs to the category of thought, model description to the category of speech and resources as well as activity to the category of things. - This dissection as such is probably new. However, similar though less systematic and less comprehensive dissections have been made before in literature, cf. 9. Jahnukainen,13-14,63,66; 10. Jahnukainen,16-17; 6. Atk-instituutin työryhmä,2-21..22; 13. Kerola-Järvinen,71.

---

1 According to the general definition of system, system is conceived as a set of interrelated elements /1. Ackoff, 662/.

2 The applicability of this dissection is not restricted to data processing systems. It can be applied to any functionalsystem, especially to any artifact.

System of data and processing

Let us first analyze data processing system as dynamic system belonging to the category of things, i.e. as data processing activity. By data processing activity we mean a sequence of behaviour performed in space-time continuum that constitutes a system and aims at the production of common output data, see 1. Ackoff,666.

Any data processing activity can be conceived as a state transition or as a sequence of state transitions. The concept of state presupposes that data processing activity is examined at an instant of time. An instantaneous examination excludes all activity from sight; only data that are to be processed can be seen. The state of data processing activity is mainly defined by data that exists at that moment. The concept of transition presupposes that data processing activity is examined at an interval of time. During this interval processing can be seen; the object of processing is data.

Traditional data processing literature emphasizes processing (transition) and the model of processing and neglects data (state) and the model of data. Yet, the goal of data processing is the production of output data and not carrying out of certain processing. The concept of processing is subordinate to the concept of data also in another sense: every unit of processing can be defined by the units of data associated to the initial and the terminal points of processing.

Data processing activity consists thus of interrelated units of data and units of processing. The purpose of examination defines whether units of data are considered as elements linked by units of processing or vice versa.

System of resources

Let us then analyze data processing system as static system belonging to the category of things, i.e. as system of data processing resources. By data processing resources we mean those material and immaterial tools or means that are necessary in data processing activity for the production of output data. The central earmark of resource is thus possibility to bring about action.

Resources can be divided into functional and non-functional resources. Functional resources are capable of processing data; therefore they can be called data processors. Individual human actor or collective human actor supported by appropriate tools can act as data processor. The knowledge that is associated to human actors may be regarded as distinct resource. Computer or computer network can also act as data processor. The systems programs that are associated to computers may be regarded as distinct resource Often human actors and computers together form data processor (man-machine system). Data processor needs a model to guide it in its data processing activity.

The general model of data processor is described in fig. 1.4. Data processor is thus an aggregate of receptors, emittors, processor and memory.

However, data processors supported by tools are not the only resources required in data processing activity. Also data mediums and premises are required. By data medium we mean substance or physical phenomenon that can be used for carrying data. Normally there can be no data without processing, i.e. data has to be generated. By means of premises we define, where data processors and data mediums are located. - Money is not a proper resource. It has value only to the extent that it can be used for bargaining proper resources. Cf. the classification of resources in Il. Järvinen-Kerola. Fig. 2.2. illustrates

the relationships that prevail between the different types of resources.

System of models

Let us next analyze data processing system as static system belonging to the category of thoughts, i.e. as system of models. We first define, what is meant by model: Object A^+ is a model of object A to subject B to the extent that B can use object A^+ to answer questions that concern object A /19. Minsky,426/. According to the general definition model could equally well belong either to the category of things, thoughts or speech. In the context of systemeering it is appropriate to interpret models as thought structures (or images) in consciousness of subject.

If data processing system is model , what is it then model of ? This question is examined from a few central points of view.

Data processing system (model) defines on one hand data processing resources and on the other hand data processing activity that will be carried out by resources, see fig. 2.1. and 2.3. The model of resources defines resources and their mutual relationships. The definition can be general, i.e. only the requirements that the individual resource has to fulfill or the desirable resource class is specified. The definition can also be individualizing, i.e. every individual resource that belongs to the system of resources is named. The model of activity defines data processing activity with respect to data processing resources; in other words the model of activity is constructed using instructions which belong to the instruction repertoire of resources. The more primitive data processing resources are (i.e. the more meagre and less powerful the instruction repertoire is) the more complex the model of data processing activity has to be in order to specify the given activity.

As data processing activity consists of units of data and units of processing, the model of data processing activity has to consist of models of data and models of processing respectively, see fig. 2.3. The model of data (data base schema, file definition) defines the object of processing; it defines, how the potentialities of data medium are used, what signs are inscribed on data medium, what interconnections there are between signs and what is meant by the signs. Although the model of data defines data at a certain instant of time, it is a model for data processor (and thus for processing), which generates, destroys or inspects that data. The model of processing (program) defines in a way which is understandable to data processor, how the data are manipulated. The models of data define the initial state and the terminal state of the state transition; the model of processing defines the state transition itself. - The basic model of data processing activity is illustrated in fig. 2.4. It identifies input data, output data , stored data and process. The model as such is not consistent with the "state transition" setting; it is consistent with the abstraction of data processor, see fig. 1.4.

It is worth noting that data processing system as model is not any objective model; it is subject's model of data processing activity and resources. Each model is constructed by using a language, which is based on a set of concepts. In the systemeering community there should be a standard set of powerful concepts as well as a set of artficial languages for defining data processing activities and resources. Besides concepts and language, also the beliefs, the values and the norms of subject can be reflected in the model. However, not all issues are equally sensitive to subject's view of the world. As the object to be modelled is complex, various models have to be used. Every model defines some aspects of the object.

Data processing system as model has a twofold nature: it is both descriptive and prescriptive. Model describes data processing resources and data processing activity (provided that resources act in accordance with model) as conceived by subject(1). Model describes, what the system of resources and activities is like. If subject(2) wants to know something about data processing resources or activity he can use the model as source of knowledge. Then model assists subject(2) in his attempt to observe real world, cf. the definition of model. Model also prescribes resources to act in a certain way; in other words it defines, what the system of resources and activities ought to be like. This prescription is subject's conception, how the world should be changed in order to reach the state of affairs indicated by his values.

System of descriptions

Let us finally analyze data processing system as static system belonging to the category of speech. By description we mean a representation of model that can be reached by the senses of subject. Interrelated descriptions constitute the system of descriptions. The descriptions of models have a few functions. Firstly, in descriptions the thoughts of subject are captured; thus they aid subject in his thinking. Secondly, in descriptions the thoughts of subject are stored for later scrutiny. It is well known, how human memory is not always to be trusted. Thirdly, descriptions are the means by which thoughts of subject can be transferred to other members of community.

Models are constructed by using an appropriate language. Descriptions of models are made by applying a mode of using language. The mode of using language should be selected so that it suits the interpreter of descriptions. Sometimes the model is translated into another language, i.e. converted into another model by using another set of concepts and another language before it is represented as

a description.

Fifteen different interpretations of data processing system

Above four basic interpretations were given to data processing system: data processing system was namely conceived as model, model description, resources or activity. By combining these basic interpretations 15 different interpretations of data processing system can be received, see fig. 2.5. The contexts in which these interpretations make sense are studied in the following chapter.

3. THE ACTIVITIES RELATED TO DATA PROCESSING SYSTEM - AN ANALYSIS OF THE ENVIRONMENT OF DATA PROCESSING SYSTEM

As a starting point for the analysis of activities related to data processing system we use the setting in which subject interacts with his environment, see fig. 1.5. Data processing system is the object in the setting. Normally some part of organization , which is interested in data processing system, is the subject in the setting. As subject is part of organization (it is an artifact), it can (if necessary) also be dissected into a system of models, model descriptions, resources and activities with respective links to data processing system. According to the setting 1) subject acts on object; 2) subject is acted on by object (this case is left unanalyzed); 3) subject observes object and communicates about object; 4) subject emits symbols to object; 5) subject receives symbols from object. Each of these cases imply one or more activities that are related to data processing system. In the following these activities are listed and discussed.

3.1. Subject acts on data processing system

A thought model on management

Subject's need for information drives him to act: to produce information. According to the classical thought model of management "management cycle" the carrying out of any task (also the production of information) can be divided into three subsequent phases: planning, execution and control. Traditionally planning, control and some tasks related to execution have been considered as managerial activities. The actual execution of the task is the activity to be managed.

When the management of artificial system is examined, artificial system (e.g. data processing system) should be interpreted as an aggregate of the static system of resources and the dynamic system of activity. Both the system of resources and the system of activities are managed. The management of resources includes planning of resources, acquisition of resources, reparation,maintenance and development of resources as well as control of resources, cf. 11. Järvinen-Kerola. The management of activity includes planning, directing and control of activity. Fig. 3.1. illustrates the model of activity and resource management as well as a combined model of management based on the two previous models. See 22. Newman.

The managerial activities can be divided into P-, S- and C-management. P-, S- and C- stand for the three main aspects of data processing system. P-management is concerned with how data processing system influences its environment; S-management is concerned with data and their interdepencies and C-management is concerned with data processing resources as well as how these resources should act in order to produce output data from input data, see 13. Kerola-Järvinen, 15-19,85.

The managerial activities related to data processing system

Fig. 3.2. represents a model of the activities , mainly managerial activities related to data processing system. This model is based on the combined model of management (fig. 3.1.) and the dissection of data processing system (fig. 2.1.). Let us now give a short discussion on each of the identified managerial activities.

Planning (design) can be dissected into two different kinds of activities: model construction and model description. In model construction an artificial systemeering language is used. The building blocks that are used are thoughts (or images) and the outcome is model, a structure of thoughts (or a structure of images). Model defines the data processing resources and the data processing activity that is to produce the desired data.

Instead of model construction one might as well talk about problem solving.[1] Systemeerer is facing the problem, why, in fact, any data processing should be carried out (P-aspect), what the data processing is that should be carried out (S-aspect) and how that data processing should be carried out (C-aspect). The problem is solved, when systemeerer has been able to answer these "why"-, "what"- and "how"- questions in his mind, i.e. when the model of resources and activity is complete. Thus model is the solution of the problem.

An essential part of planning is model description i.e. documentation. There is not much use of the model (the problem solution) unless systemeerer communicates it to others, whom it concerns. The mode of description depends on the interpreter. Man and computer differ

---

[1] Still another alternative is to talk about decision making, the outcome of which is a decision.

significantly in their ability to interpret model descriptions. The separation of model description from model construction is mainly analytical. In practice they should proceed intertwined. - On the completion of planning data processing system works only on paper.

Resource acquisition may be started when the planning of resources has proceeded far enough. The skills needed in resource acquisition are different from the skills required in planning. The outcome of resource acquisition is system of data processing resources which is prepared for data processing activity. The supplementation of short term resources is thought to belong to the direction of data processing activity.

The construction of data processing system is a managerial activity concerning data processing resources and activity. It consists of planning (model construction + model description) and resource acquisition. The outcome of construction process is data processing system, which is prepared for action.

The so called second installation of data processing system /15. Kuva-projekti/ is identifiable with the latter part of construction process. The point of departure for second installation is a fairly complete plan of resources and activity. This plan has perhaps deliberately been made so general that it can be applied to many situations (generalized programs and generalized data processing systems). The purpose of second installation is to complete the plan and acquire the necessary resources.

From the point of view of direction data processing system is a system of resources and activity. Direction is necessary among other things because plans can not be very detailed and because execution of plans is most often affected by unexpected events. In direction we include the acquisition of short term resources, allocation of

resources to tasks, timing of activity and control of error free work flow, cf. 15. Kuva-projekti. If organization has generalized programs and is able to link several computer runs together, quite big computer runs can be performed guided only by control files. The planning and control of the contents of these control files is part of direction.

Although control focuses the attention to the past, the purpose of it is the improvement of the future system of resources and activity. Following the division of management control can be divided into P-control, S-control and C-control. P-control is concerned with the meaningfulness of data processing activity; it observes the development in the system environment and the influence of data processing system upon its environment. S-control observes the gross value of output data; S-control is interested in, whether end user receives according to plans the information that he wanted. C-control is concerned with the shape of resources and the efficiency of activity. See 18. Leppänen, et.al., 124-135.

By changing of data processing system we mean construction of data processing system that is initiated by the feedback from control. Referring to management cycle, changing is the first phase in the n'th management cycle ($n \geq 2$). Various types of changing activities can be identified, see 15. Kuva-projekti,32; they are discussed in short in the following.

By reparation of data processing system we mean the corrective actions that make the defective resources and the malfunctioning activity obey the original intentions. The need for reparation arises from errors in plans (e.g. in programs) and from decay of resources. - Sometimes replacement of resources is necessary; this does not normally affect the plans. - On other occasions the plans for activity and resources have to be changed and

consequently changes inthe system of resources are induced. Based on the three main aspects of data processing system we can distinguish maintenance and development of data processing system. By maintenance we mean activities that change data processing resources and the way in which they produce the desired output data (C-aspect). By development we mean the activities that induce changes in the desired influence of data processing system as well as in data and their interdependencies (P- and S-aspect).

To give some perspective to the managerial activities that guide data processing system we give a model of organization, and two models of management hierachies. The model of organization gives the context of data processing systems (see fig. 3.3; cf. 11. Järvinen-Kerola). The first model of management hierachy represents the management of the entire corpus of data processing systems in organization (see fig. 3.4.), whereas the second one represents the managerial activities that influence a single data processing system (see fig. 3.5.).

3.2. Subject observes data processing system, subject communicates with data processing system

If subject is interested in data processing system he has in principle two alternatives in getting acquainted with data processing system : a) subject has to observe data processing resources and activities himself, or b) subject has to rely on the model of data processing resources and activities formed and communicated by somebody else. Alternative b) is more economical and more available and thus more common : one becomes acquainted with data processing system through the presentation of model descriptions. The most important group to whom data processing system is presented is the present and potential members of interest groups of data processing system, such as end users, systems analysts, programmers. Some data processing systems have to be presented to authorities and

organization's internal auditors. Data processing systems are also presented to students (education), researchers (research), colleagues, guests and sometimes to the public (mass communication, pr).

When the communication of data processing system is examined data processing system is interpreted as activity. Every data processing system is preceded either by another data processing system or by data acquisition and is succeeded by another data processing system or data utilization. By data acquisition we mean activity, where data is generated. Data acquisition is based either on observation of real world, reception of symbols, process of inference or inspection of memory, cf. chap. 1. By data utilization we mean activity, where end user interprets data and lets it influence his behaviour.

Heikki Kämäräinen

REFERENCES

1. Ackoff, Russel — Towards a system of systems concepts. Management Science, Vol. 17, No 11, July 1971.

2. Ahmavaara, Yrjö — Yhteiskuntakybernetiikka. Weilin & Göös, Tapiola, 1976. (in Finnish)

3. Alston, W.P. — Language. In Edwards, Paul (ed.), The Encyclopedia of Philosophy, Collier and Macmillan Ltd., London, 1967.

4. Alston, W.P. — Meaning. In Edwards, Paul (ed.), The Encyclopedia of Philosophy, Collier and Macmillan Ltd., London, 1967.

5. Alston, W.P. — Sign and symbol. In Edwards, Paul (ed.) The Encyclopedia of Philosophy, Collier and Macmillan Ltd., London, 1967.

6. Atk-instituutin työryhmä — Tietosysteemin rakentaminen. Tietojenkäsittelyliitto ry:n julkaisu 25, toinen korjattu painos, 1975. (in Finnish)

7. Bunge, Mario — Scientific Research I, The Search for System. Springer Verlag, 1967.

8. Heath, P.L. — Concept. In Edwards, Paul (ed.), The Encyclopedia of Philosophy, Collier and Macmillan Ltd., London, 1967.

9. Jahnukainen, Miikka — Systeemimalli yrityksen informaatiosysteemin ja sen suunnittelun kehysmallina. Tampereen yliopiston tutkimuslaitos, 41, 1967. (in Finnish)

10. Jahnukainen, Miikka — Yrityksen informaatiosysteemin suunnittelun kehysmetodi. Kansantaloudellinen yhdistys, 1971. (in Finnish)

11. Järvinen, Pertti
Kerola, Pentti — Notes on research in systemeering. Studies in honour of Olavi Bertel Hellman on the occasion of his fiftieth birthday March 5, 1978. Turun yliopiston julkaisuja, sarja A,1978.

12. Kangassalo, Hannu — Conceptual structure of the file design problem: the general view. Difo-tutkimus ry, University of Tampere, Difo report 8, 1975.

13. Kerola, Pentti
Järvinen, Pertti — Systemointi II. Gaudeamus, 1975. (in Finnish)

14. Klir, George — An Approach to General Systems Theory. Van Nostrand Reinhold Co., New York, 1969.

15. Kuva-projekti — Tietosysteemin kuvausjärjestelmä. Tietojenkäsittelyliitto ry, Kuva-projektin väliraportti, 1977. (in Finnish)

16. Kämäräinen, Heikki — Tietosysteemi ja abstraktiotasoperiaate sen rakentamisessa ja kuvaamisessa. Tampereen yliopisto, Matemaattisten tieteiden laitos, raportti A-8, 1977. (in Finnish)

17. Langefors, Börje — Information and data. In Ralston, Anthony & Meek, Chester (eds.), Encyclopedia of Computer Science. Petrocelli/Charter, New York, 1976.

18. Leppänen, Mauri; Järvinen, Pertti; Kerola, Pentti — Johdatus tietojenkäsittelyyn. Tietojärjestelmien hyväksikäytön näkökulma. Tietojenkäsittelyliitto ry:n julkaisu 37, 1977. (in Finnish)

19. Minsky, Marvin — Matter, mind and models. In Minsky, Marvin (ed.), Semantic Information Processing, The MIT Press, 1968.

20. Nauta, Doede — The Meaning of Information. Mouton, The Hague, 1972.

21. Nordenstreng, Kaarle — Tiedotusoppi. Otava, Keuruu, 1975. (in Finnish)

22. Newman, William — Administrative Action. The Techniques of Organization and Management. Prentice-Hall Inc., Englewood Cliffs, N.J., 1963.

23. Searle, J.R. — Proper names and descriptions. In Edwards, Paul (ed.), The Encyclopedia of Philosophy, Collier and Macmillan Ltd., London, 1967.

24. Woozley, A.D. — Universals. In Edwards, Paul (ed.), The Encyclopedia of Philosophy, Collier and Macmillan Ltd., London, 1967.

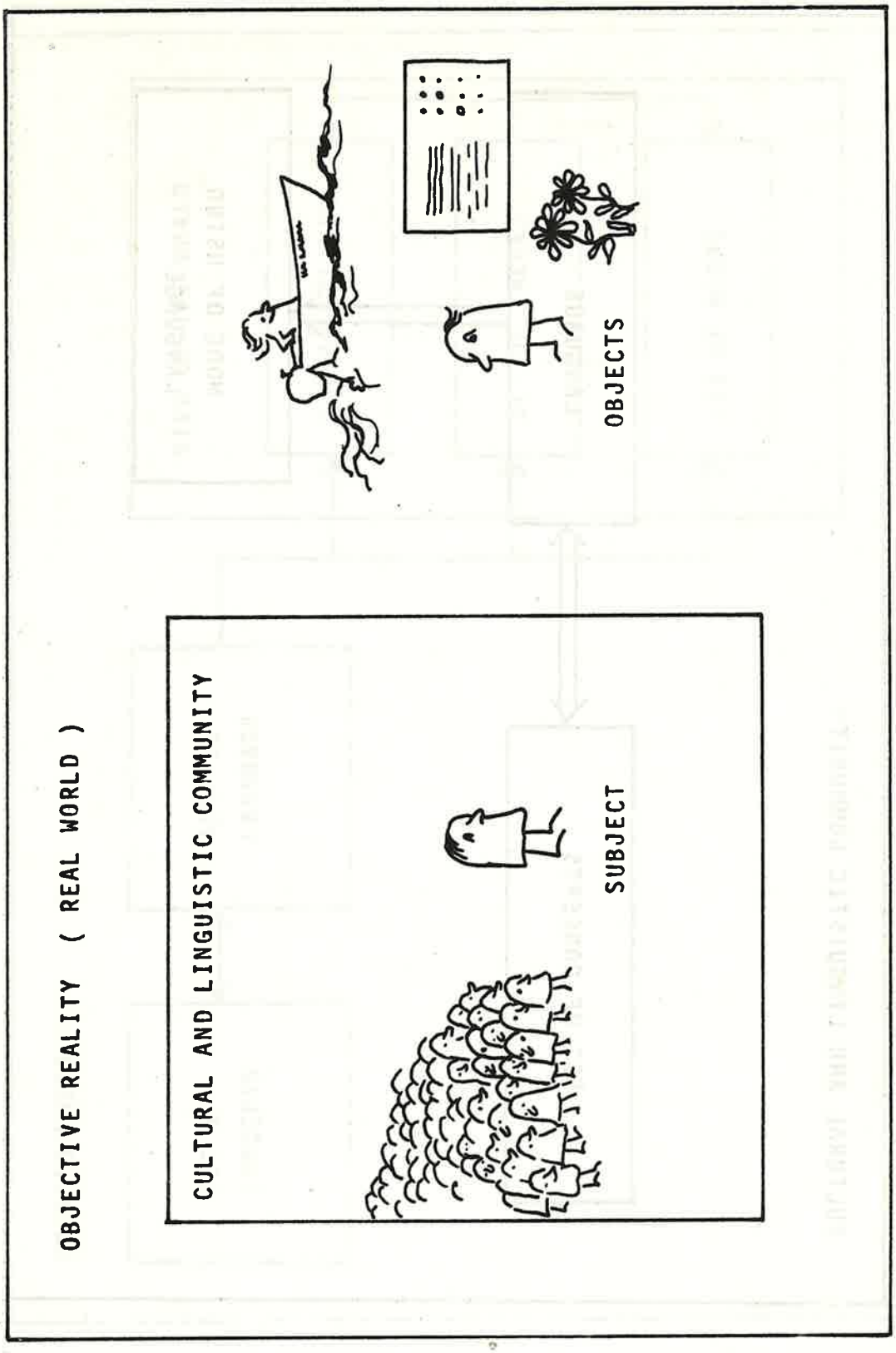


FIGURE 1.1. SUBJECT - OBJECT SETTING

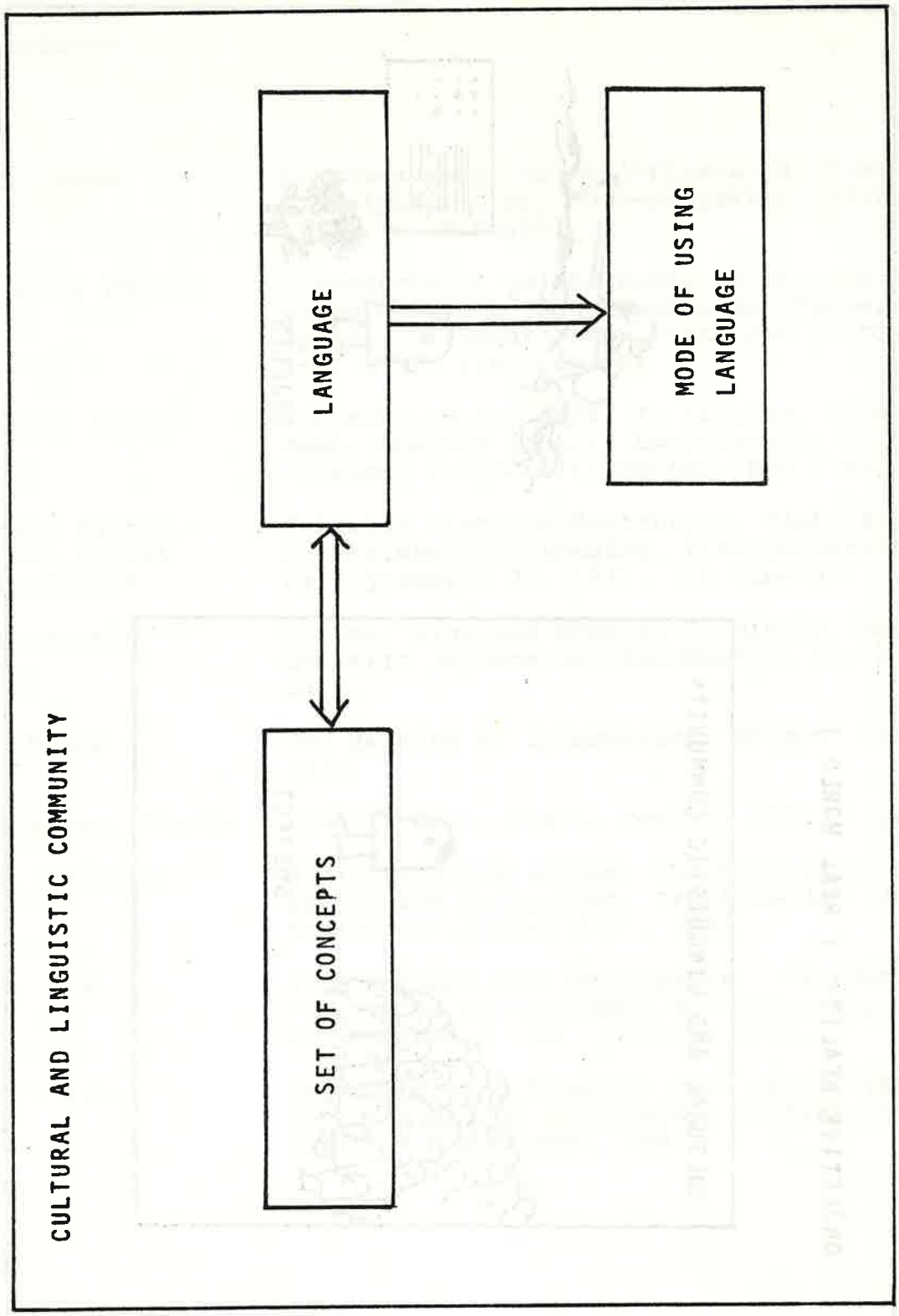


FIGURE 1.2. POSSESSIONS OF CULTURAL AND LINGUISTIC COMMUNITY

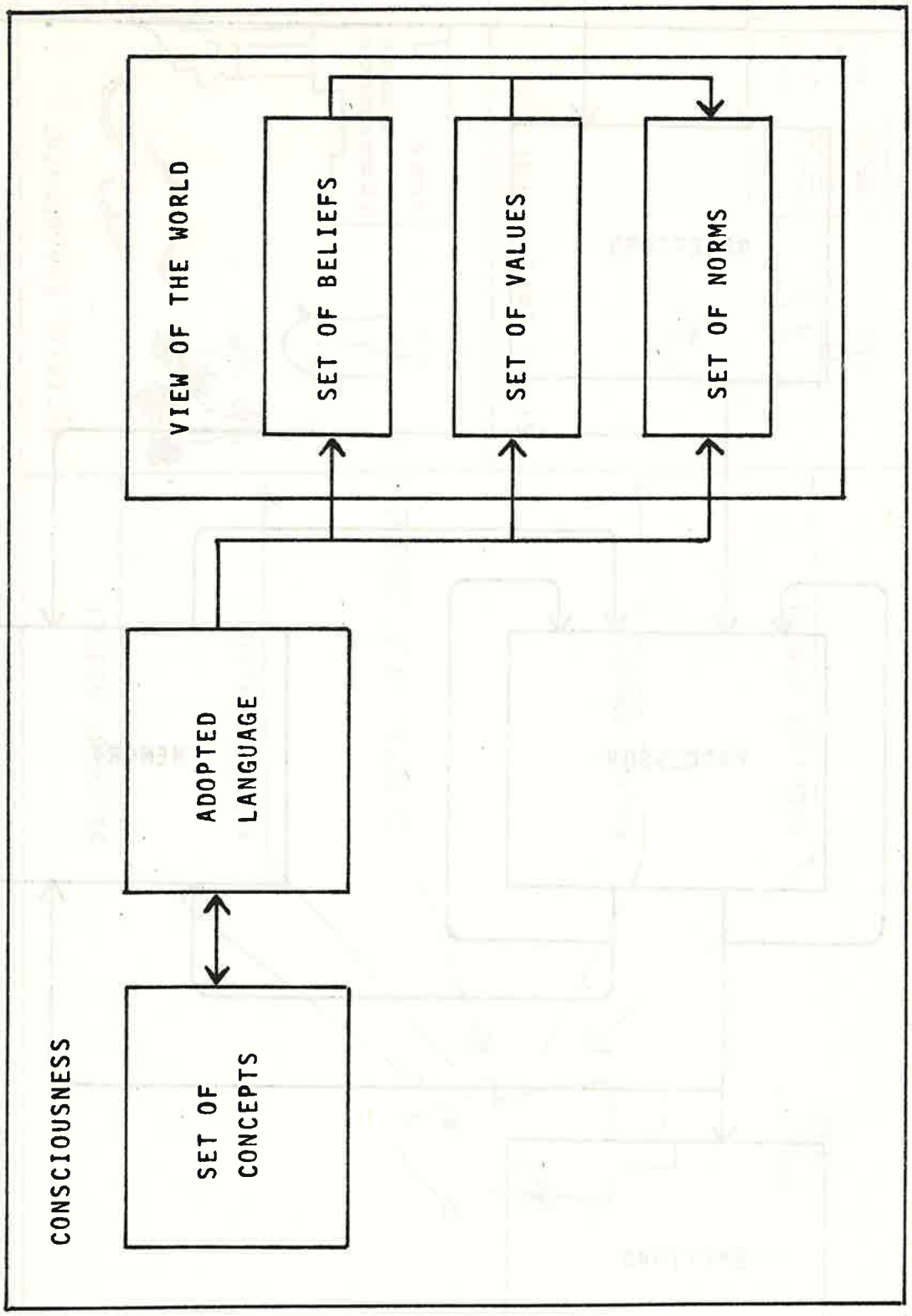


FIGURE 1.3. CONTENTS OF SUBJECT'S CONSCIOUSNESS

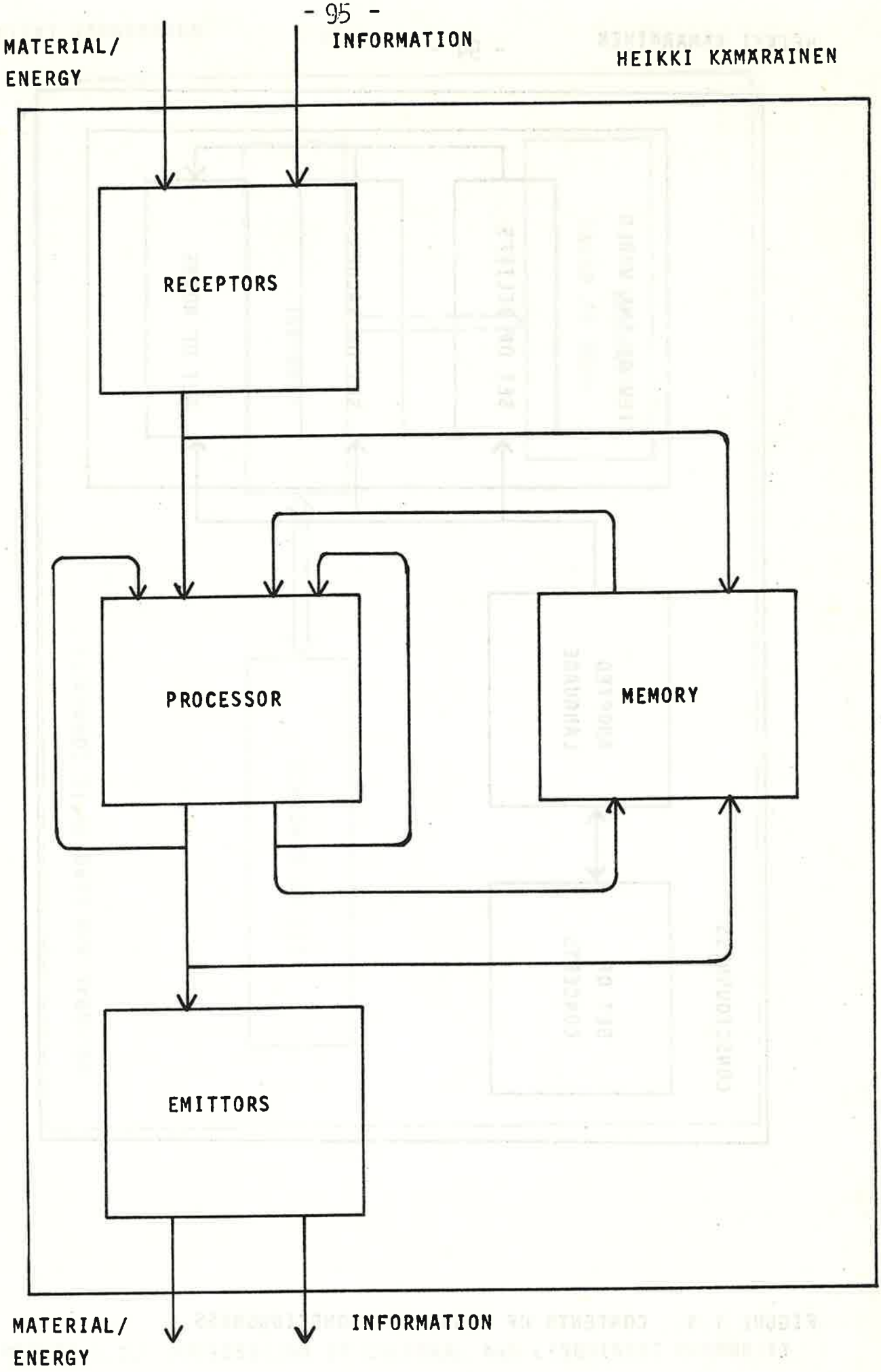


FIGURE 1.4. AN ABSTRACTION OF SUBJECT

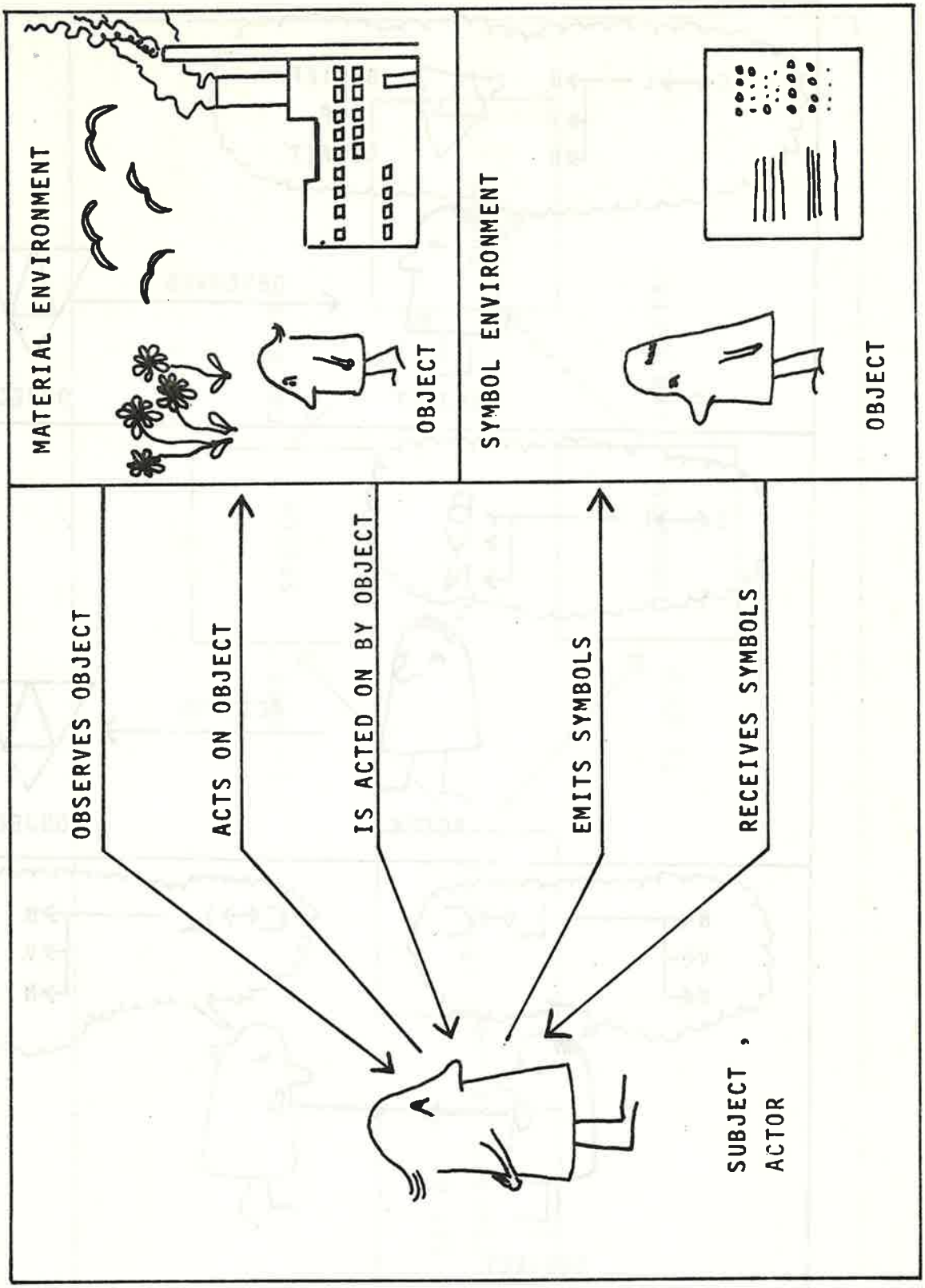


FIGURE 1.5. SUBJECT AND HIS INTERACTIONS WITH ENVIRONMENT

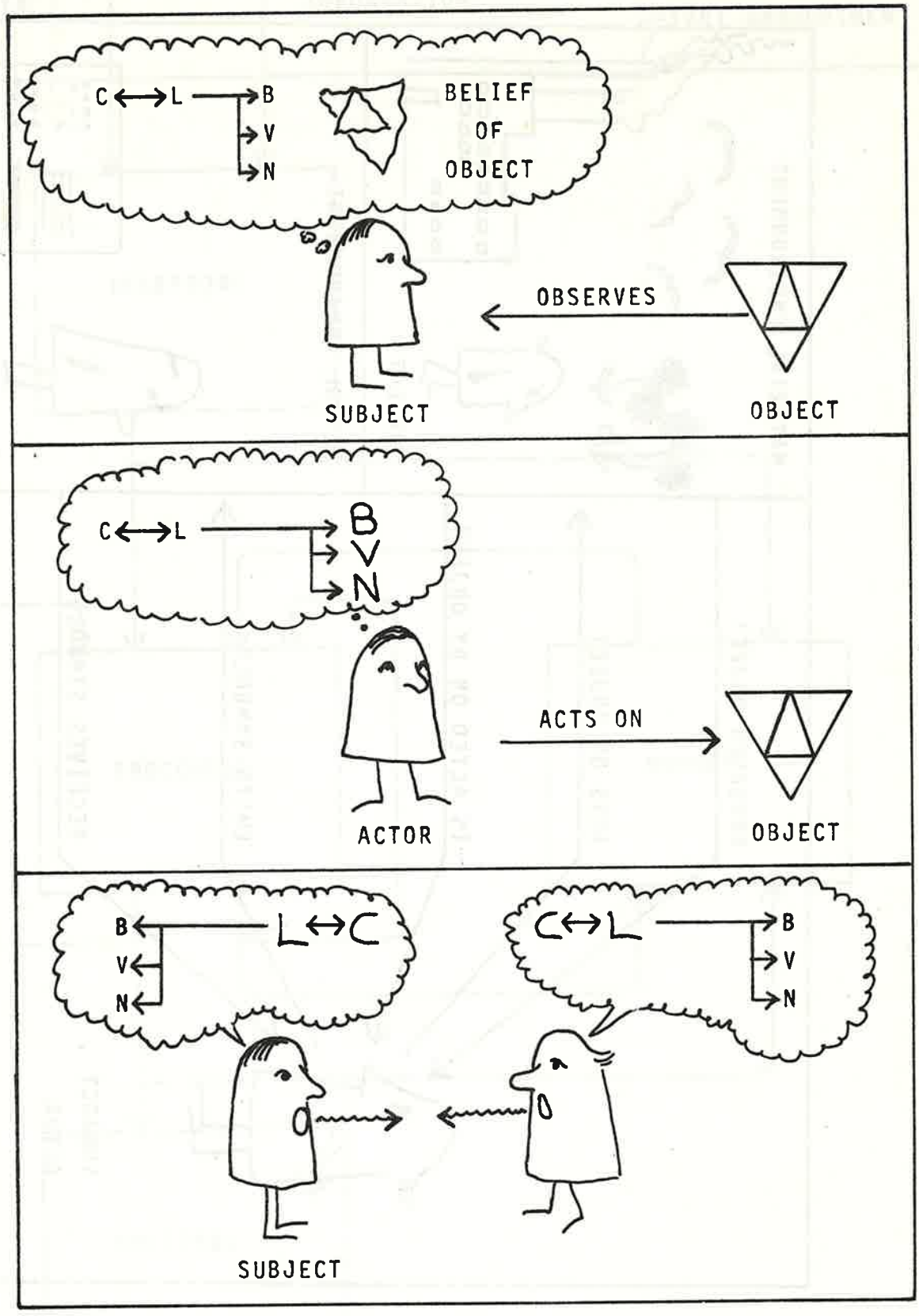


FIGURE 1.6. THE MODELS OF OBSERVING, ACTING AND COMMUNICATING SUBJECT
(C= SET OF CONCEPTS , L= LANGUAGE ,
B= SET OF BELIEFS , V= SET OF VALUES ,
N= SET OF NORMS)

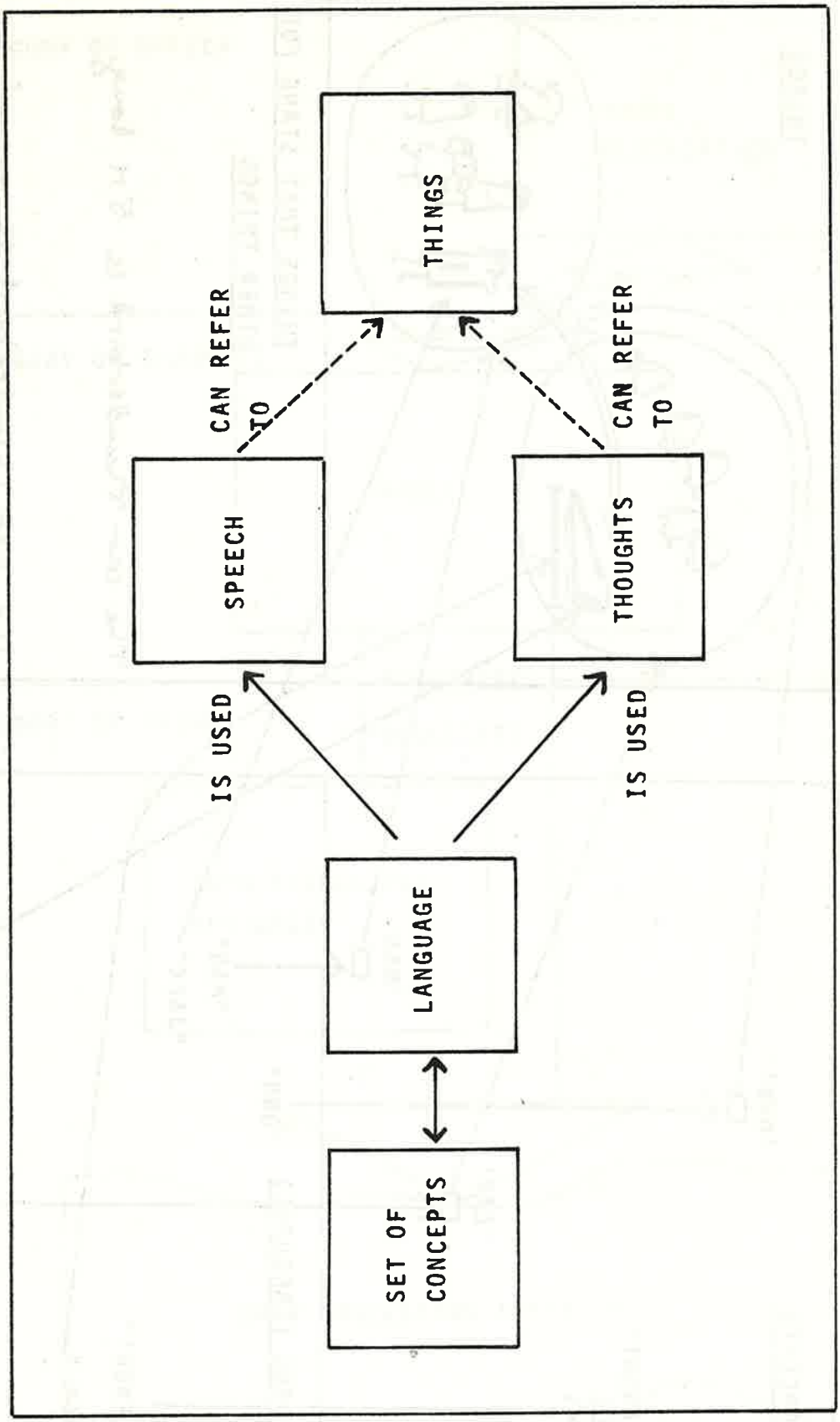


FIGURE1.7. THE BASIC CATEGORIES OF SUBJECT'S (LINGUISTIC) BEHAVIOUR

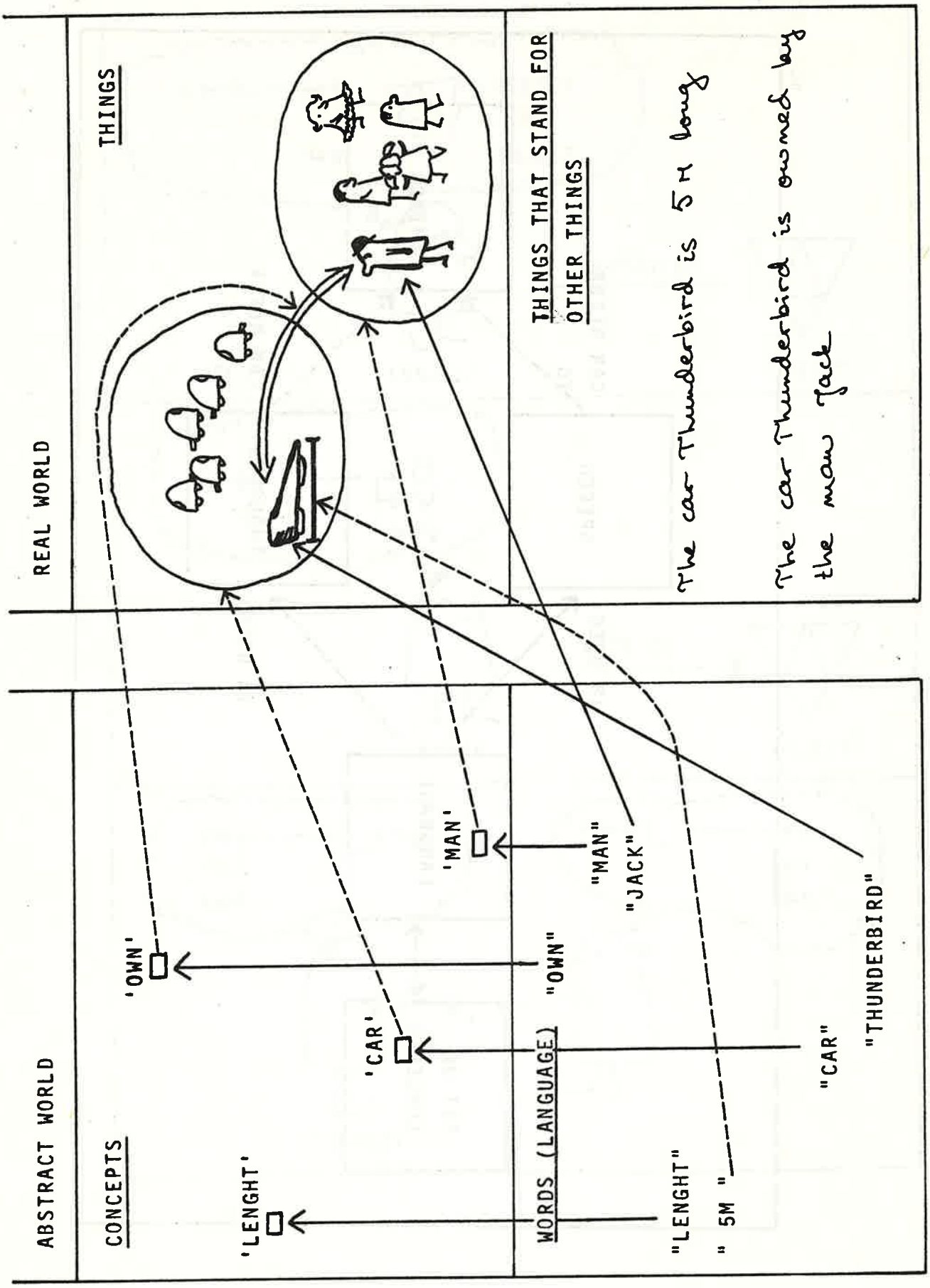


FIGURE 1.8. SOME OF THE RELATIONSHIPS BETWEEN THE BASIC CATEGORIES OF SUBJECT'S LINGUISTIC BEHAVIOUR

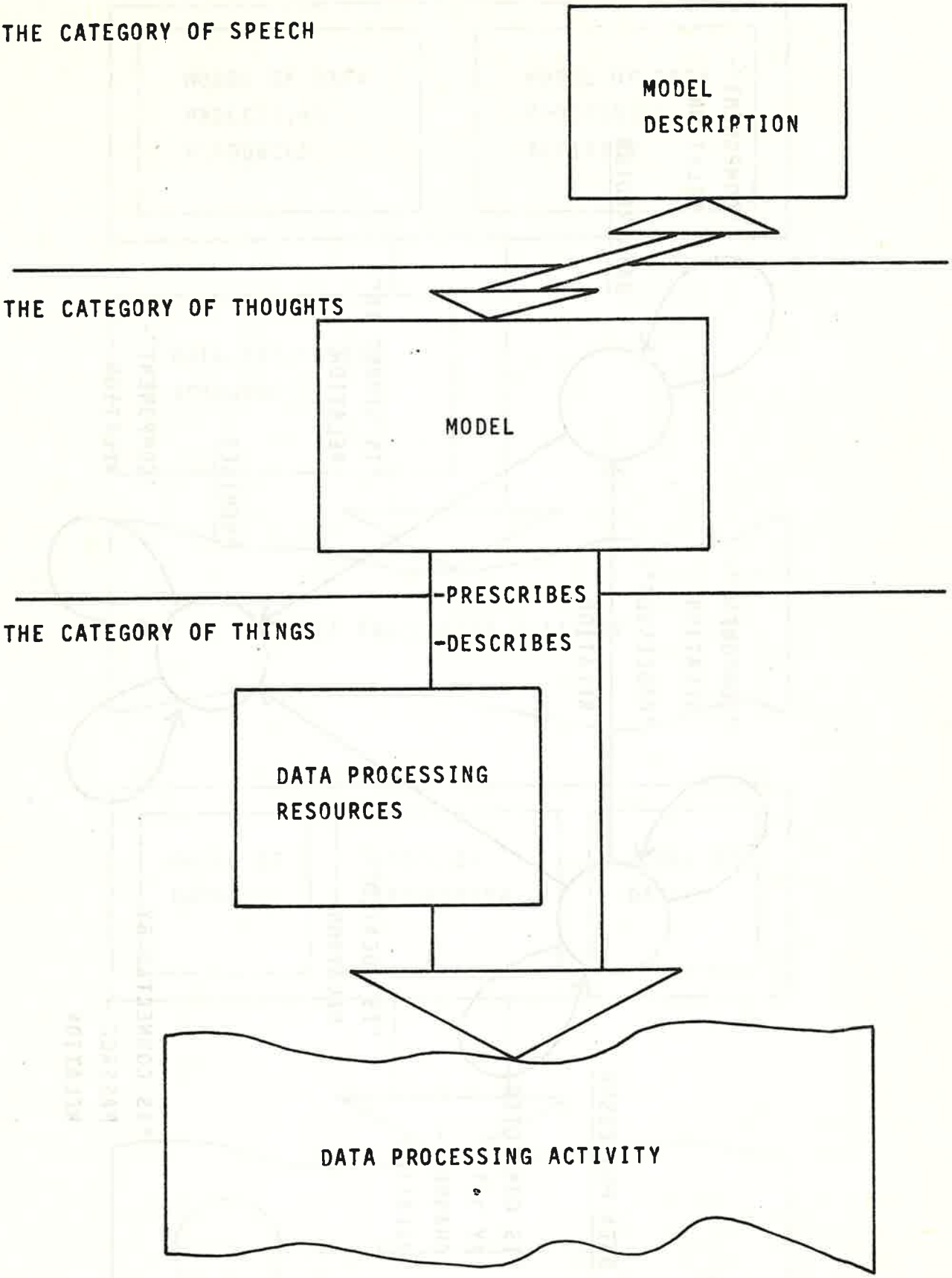


FIGURE 2.1. THE DISSECTION OF DATA PROCESSING SYSTEM INTO FOUR KINDS OF SYSTEMS

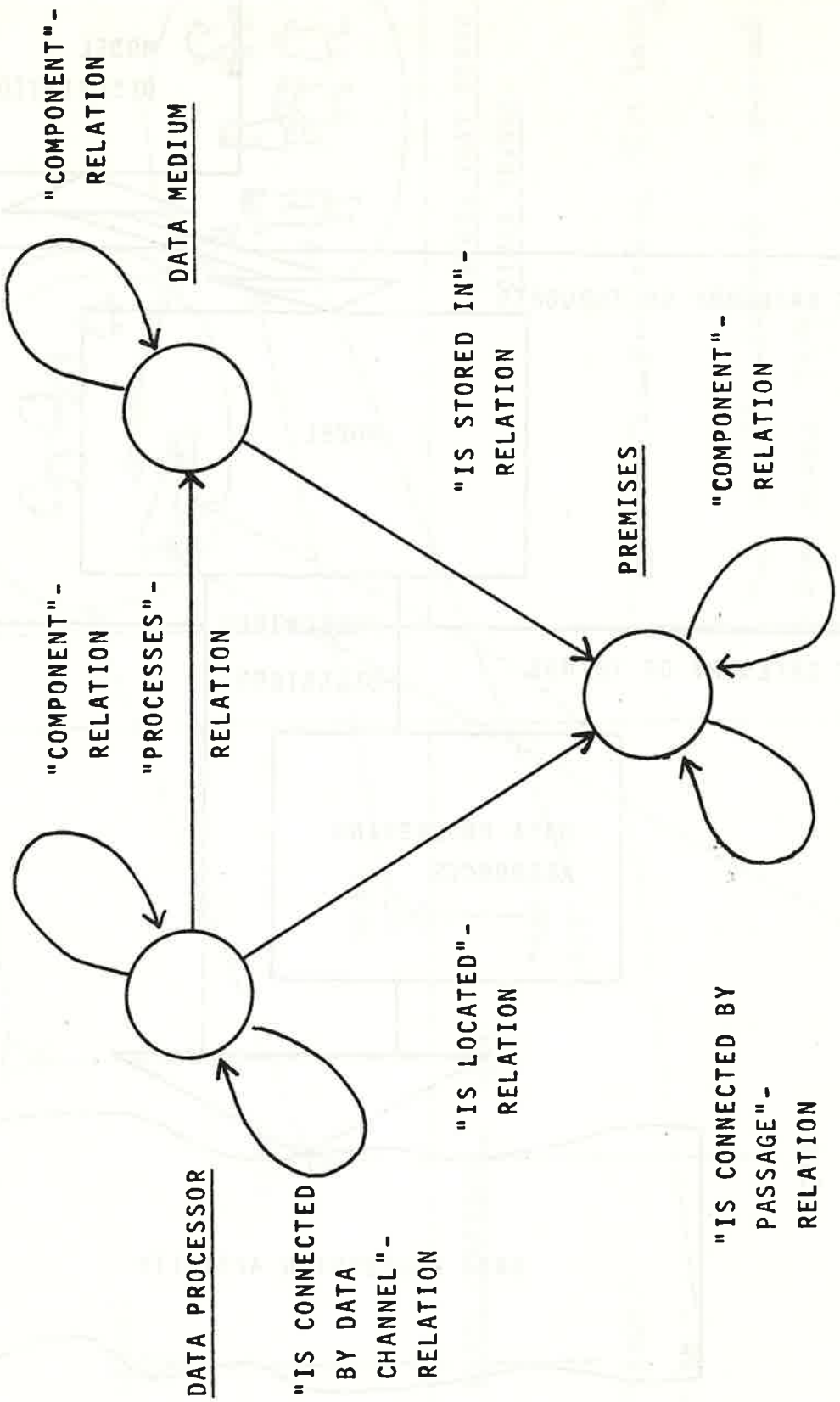


FIGURE 2.2. RESOURCE TYPES AND THEIR INTERRELATIONS

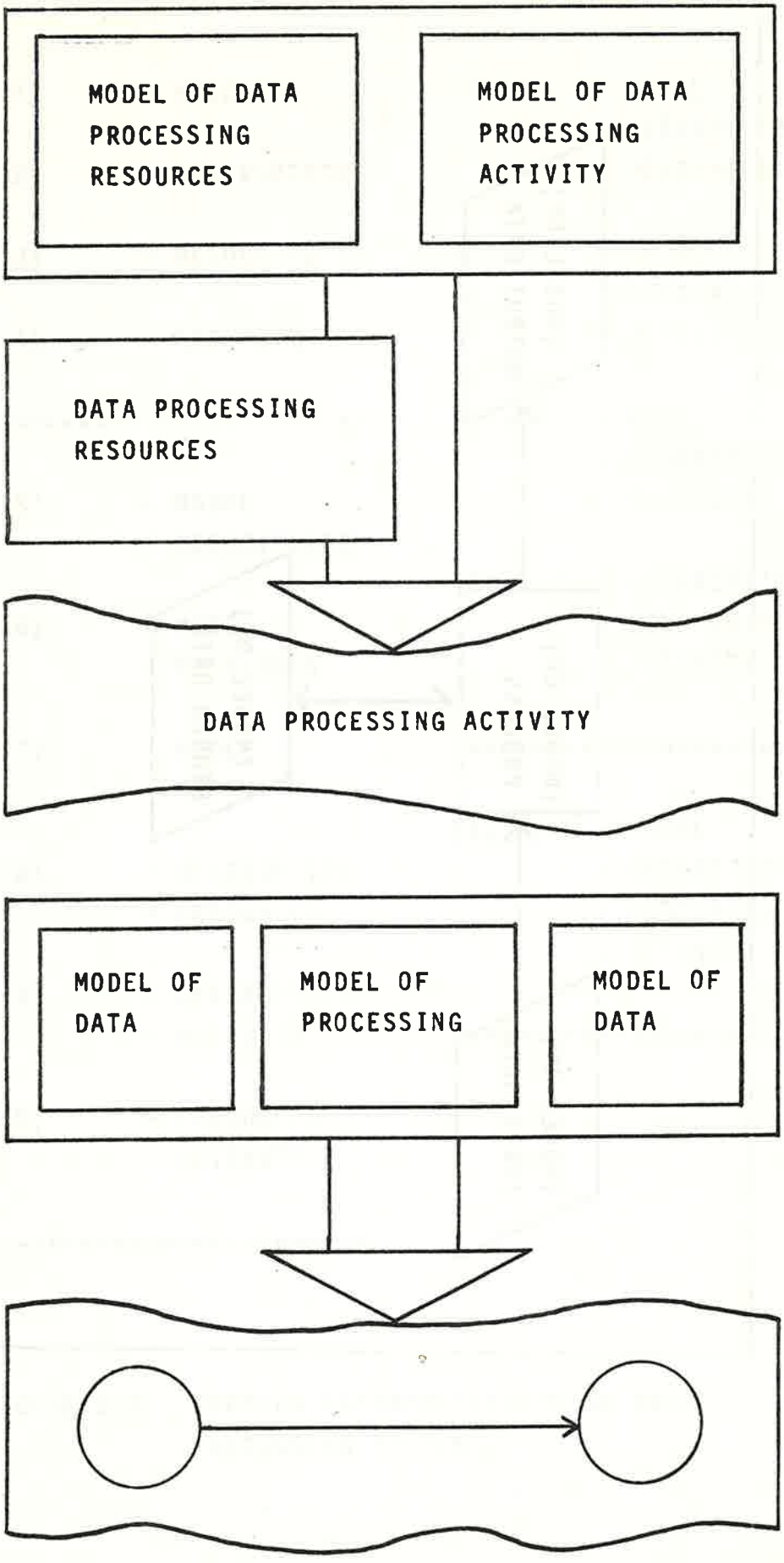


FIGURE 2.3. DATA PROCESSING SYSTEM AS MODEL

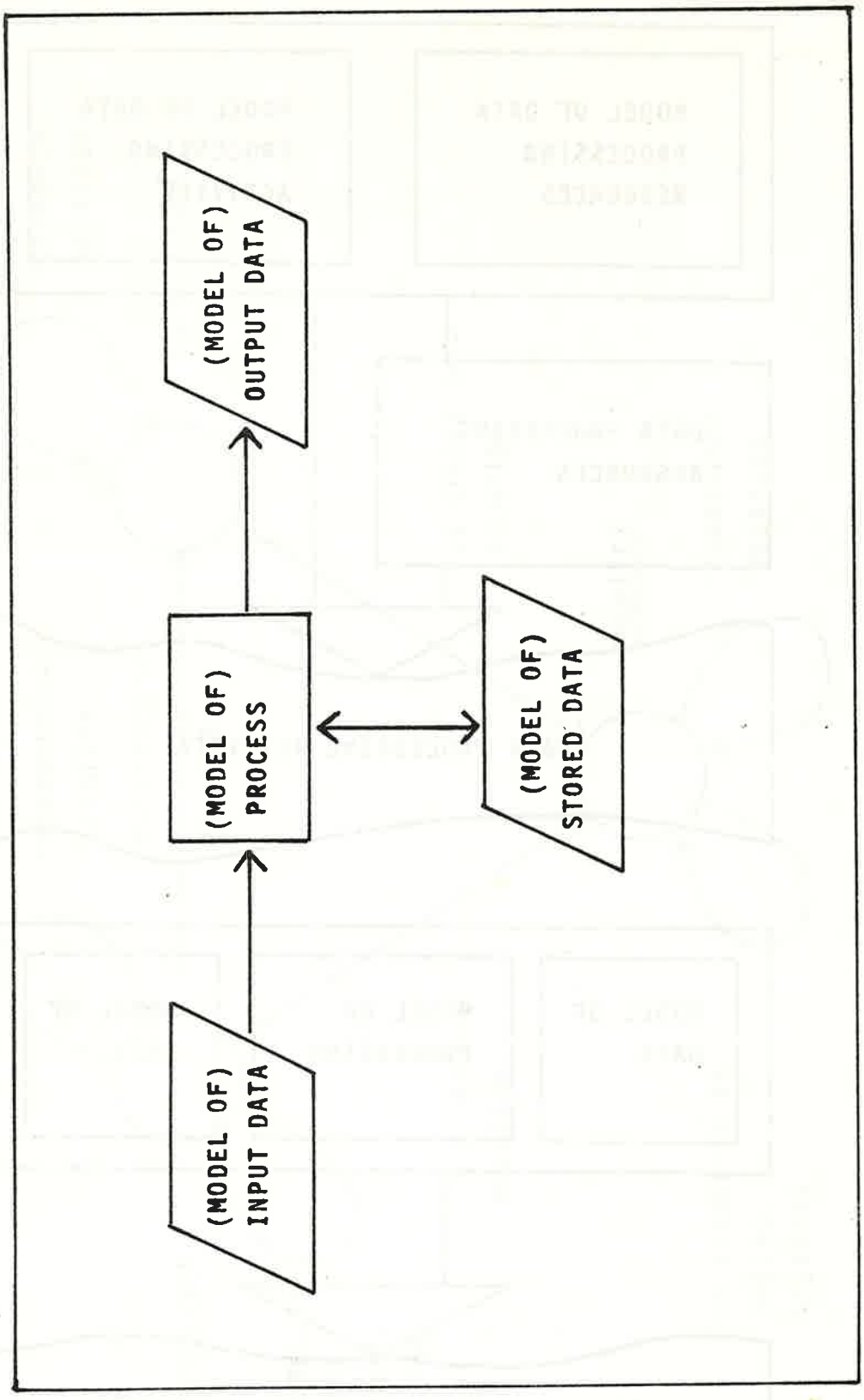


FIGURE 2.4. THE BASIC MODEL OF DATA PROCESSING ACTIVITY

1) - MODEL

2) - DESCRIPTION

3) - RESOURCES

4) - ACTIVITY

==========

5) - MODEL
- DESCRIPTION

6) - MODEL
- RESOURCES

7) - MODEL
- ACTIVITY

8) - DESCRIPTION
- RESOURCES

9) - DESCRIPTION
- ACTIVITY

10) - RESOURCES
- ACTIVITY

==========

11) - MODEL
- DESCRIPTION
- RESOURCES

12) - MODEL
- RESOURCES
- ACTIVITY

13) - MODEL
- DESCRIPTION
- ACTIVITY

14) - DESCRIPTION
- RESOURCES
- ACTIVITY

==========

15) - MODEL
- DESCRIPTION
- RESOURCES
- ACTIVITY

==========

FIGURE 2.5. FIFTEEN INTERPRETATIONS OF DATA PROCESSING SYSTEM

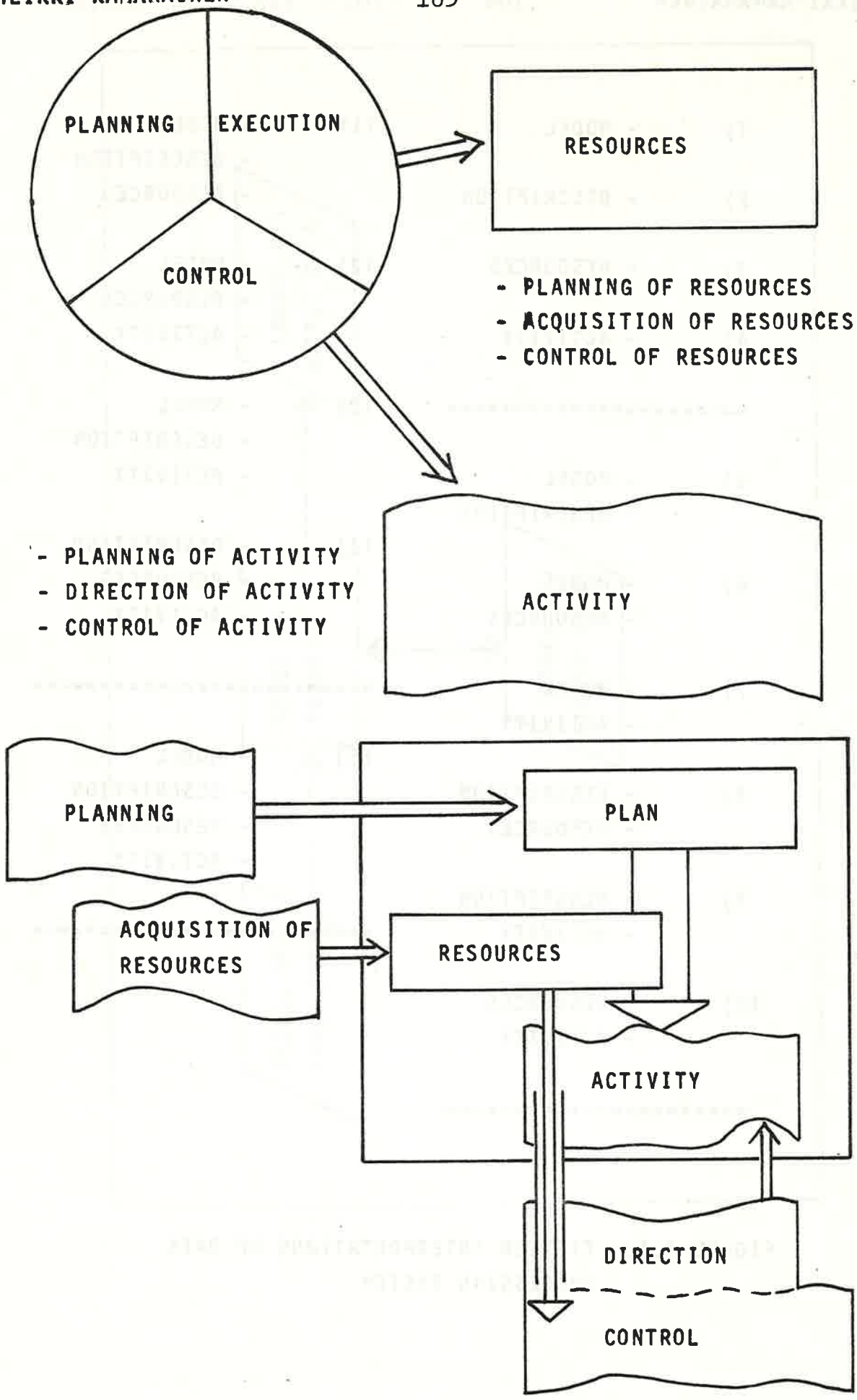


FIGURE 3.1. A THOUGHT MODEL ON MANAGEMENT

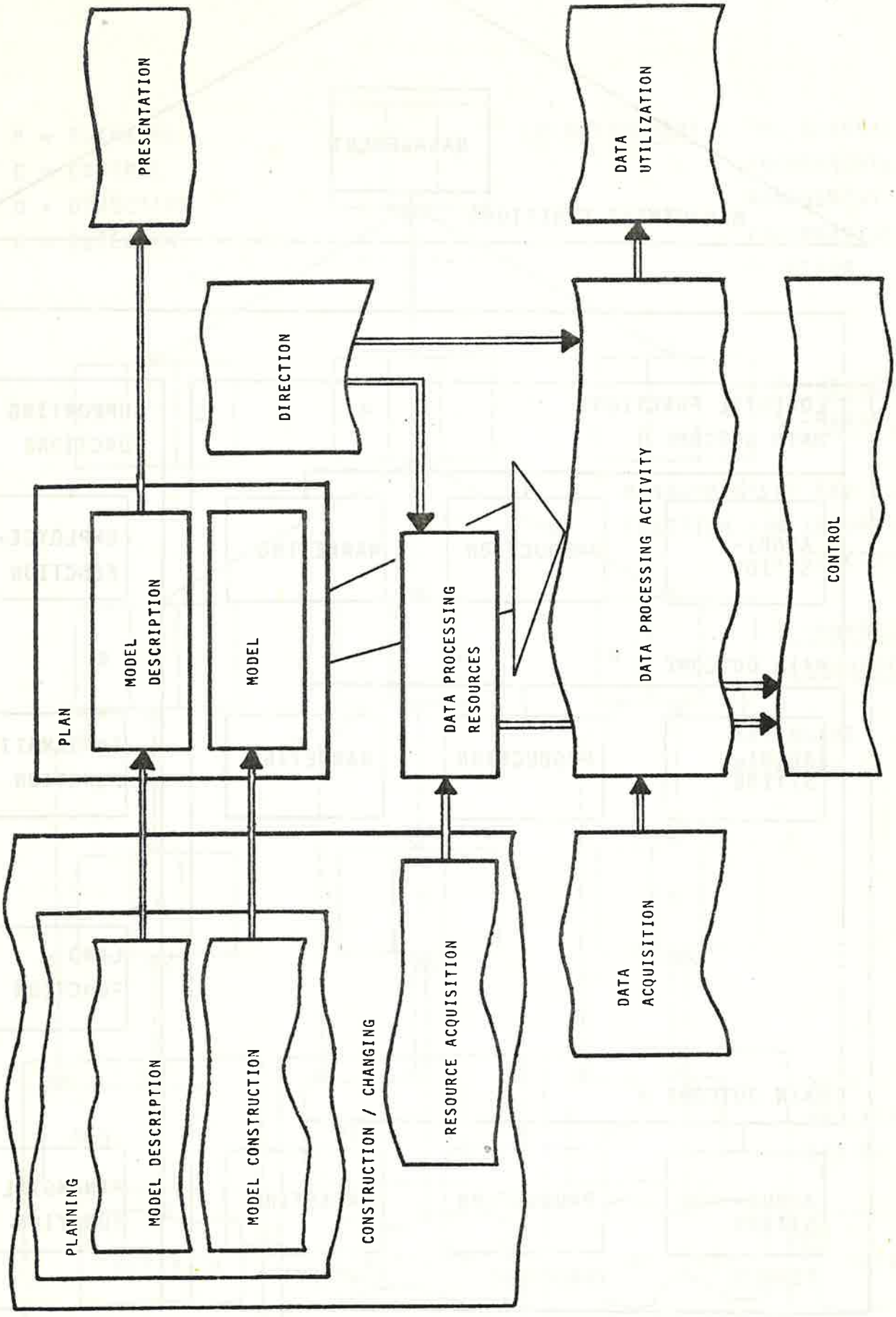


FIGURE 3.2. THE ACTIVITIES RELATED TO DATA PROCESSING SYSTEM

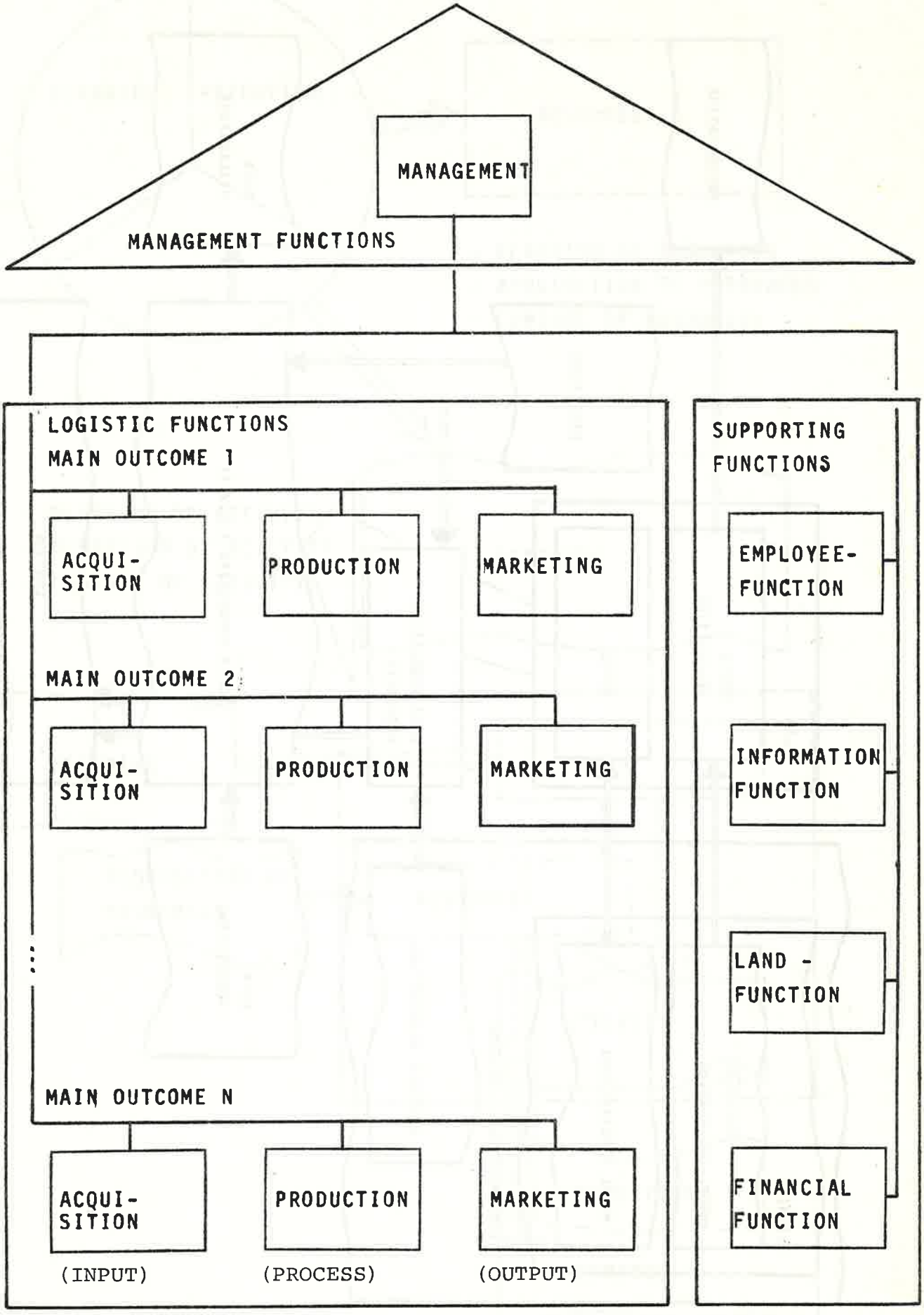


FIGURE 3.3. A MODEL OF ORGANIZATION

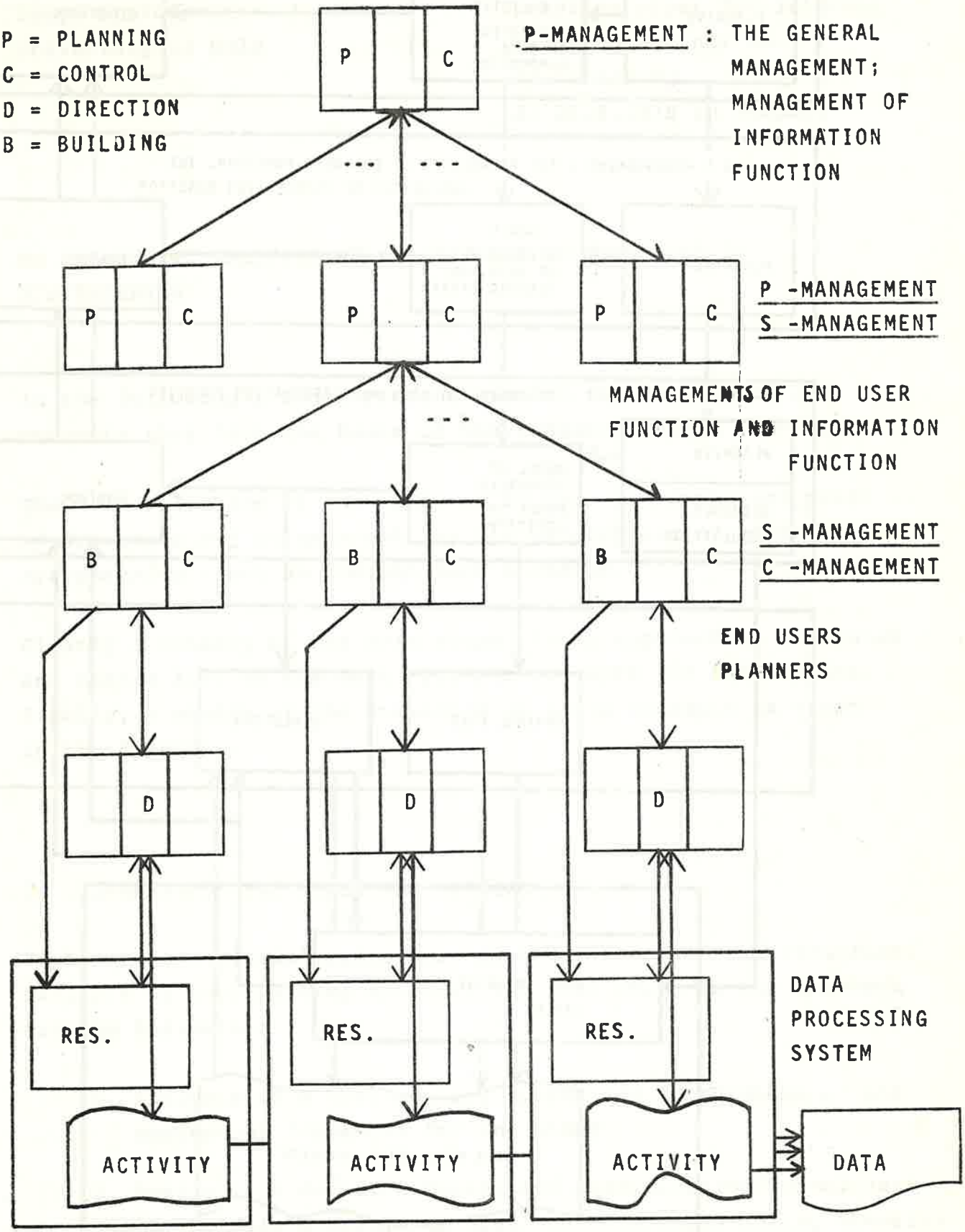


FIGURE 3.4. THE MANAGEMENT HIERARCHY OF DATA PROCESSING SYSTEMS

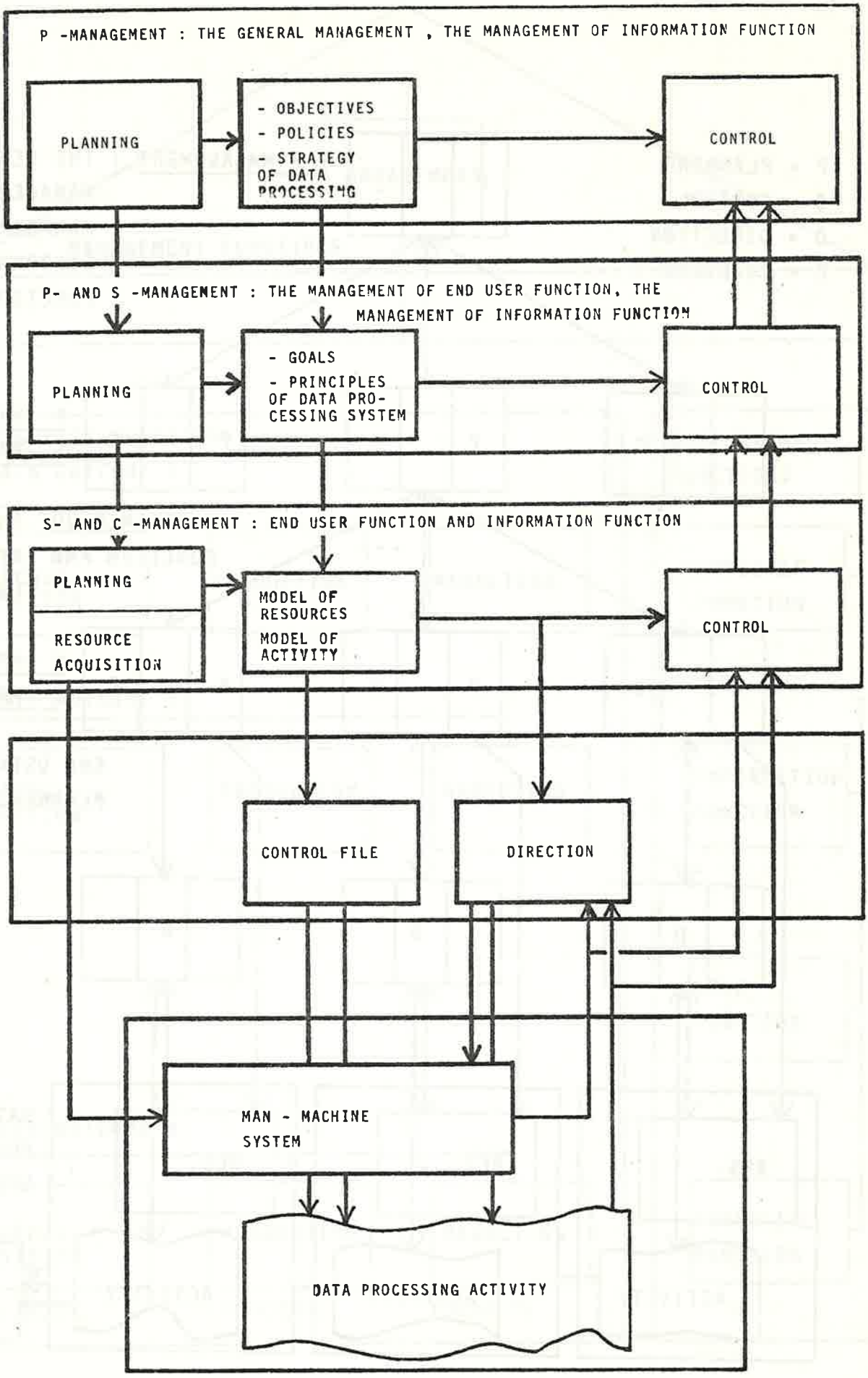


FIGURE 3.5. THE MANAGEMENT HIERARCHY OF A DATA PROCESSING SYSTEM

Seppo Mustonen
University of Oulu

Drafting paper for internal research seminar on systemeering
21-24.8.1978 in Tampere

ON CYBERNETIC ANALYSIS AS A BASIS FOR METATHEORIES IN SYSTEMEERING

In the following I shall present some of the main ideas and concepts that form the basis of the research report (MU-78).

The main objective of the study is to *show one important point of departure and an approach for the development of theories in systemeering* - not to present such theories.

Firstly a summary of the main steps of the approach is presented and then a list of the most relevant concepts for our seminar. Finally, a sketch of the structure of those concepts is given in the Appendix.

1 SUMMARY OF THE GOALS AND APPROACH

"Cybernetic analysis" means here *formal conceptual and structural analysis of goal-seeking systems*. The main steps in the approach are the following:

1. Analysis of a goal-seeking system and formulation of the conceptual framework for the study
2. Analysis of decision-making and clarification/formalization of the main problems in developing decision-making theories

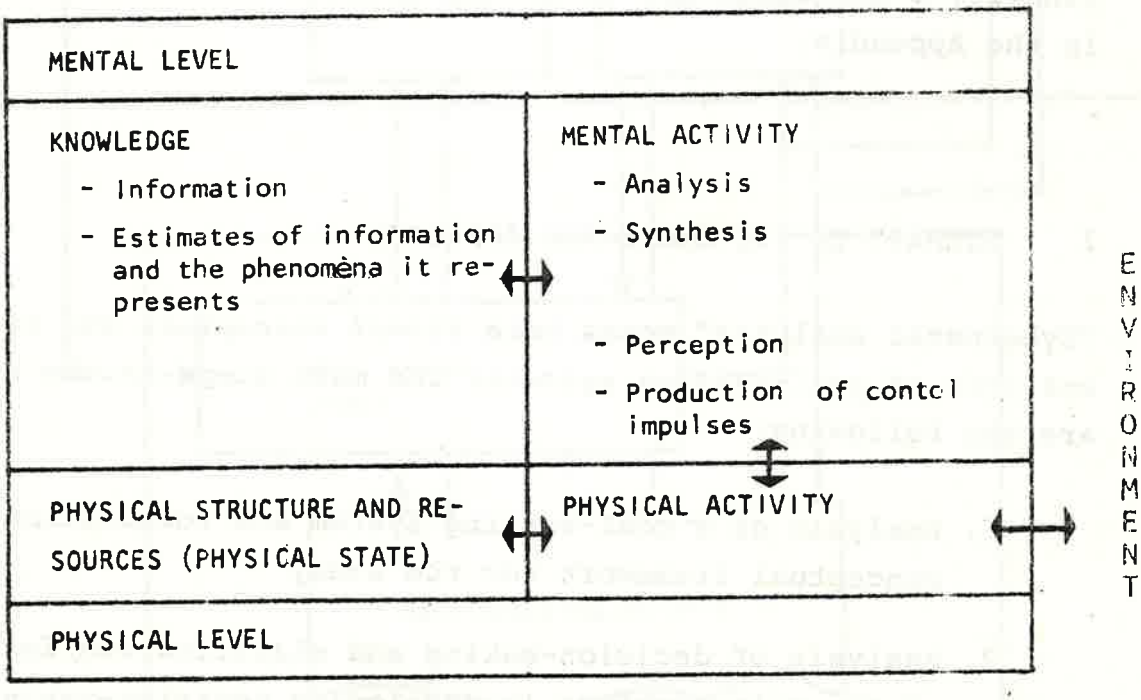
3. Analysis of systemeering and clarification/formalization of the main problems in developing theories of systemeering

In the following, each step is considered to the goals and manner of proceeding.

Analysis of a Goal-Seeking System and Formulation of the Conceptual Framework for the Study

The conceptual framework is composed of a *conceptual model of a goal-seeking system*, *an analysis of the concept of a subject* and *an analysis of knowledge*.

For the purpose of the conceptual model of a goal-seeking system, reality is divided into *mental* and *physical* levels, and at the same time, the model is also based on the *static/dynamic nature* of the properties of a system. The model is illustrated in the following figure.



The conceptual model of a goal-seeking system is formalized in chapter II.

For the analysis of conception both the static and dynamic features of conception are recognized. The static features are concerned with knowledge and the dynamic features with mental activity. Furthermore, knowledge is divided into its subcategories: *information* (which is composed of system models and outlines the real world), *beliefs*, *values* and *norms*, ie. *concept structures that estimate the truth of the models and the value/possibility of the phenomena presented by the models.*

In the analysis of knowledge and information the classification of properties into *syntactic*, *semantic* and *pragmatic* ones is applied. The presentation is based on the recognition/formalization of the most important *forms in which hierarchies are manifest* and the *main aspects (points of view) of decision-making and systemeering.*

Analysis of Decision-Making and Clarification/Formalization of the Main Problems in Developing Decision-making Theories

As a point of departure for this step we consider goal-seeking behaviour in principle and try to clarify the nature and role of *value* and *norm systems* in decision-making. The concept of *objective value* is also characterized in terms of *subjective need* and the potential for the scientific study of values and norms.

The structure of decision-making and the way in which the decision-making process is reflected in knowledge must then be considered. The analysis is based on the description of a *momentary situation of choice* as a problem of allocating one's resources to mental and physical activities. The problem of *programmability/non-programmability of decision-making* is also considered and a criterion for solving the problem is presented.

Finally, an attempt is made to clarify the main problems of the development of decision-making theories. Decision-making is considered as a problem-solving process and the *verification of the logicality of the decision-making model* is formulated as a logical testing of the validity of a hierarchical problem-solution model. Decision-making is also considered as a learning process and the *verification of the efficiency of decision-making models* is formulated on the basis of Yakowitz's theory of adaptive control processes (YA-69)).

Analysis of Systemeering and Clarification/Formalization of the Main Problems in Developing Theories of Systemeering

The concept structure obtained as a result of the structural analysis of decision-making are applied to the analysis of systemeering. The systemeering process is defined on the one hand as a problem-solving process from which the *verification of the logicality* of a systemeering model can be formalized, and on the other hand as a learning process from which the *verification of the efficiency* is formalizable.

The verification problems of a systemeering model are defined *independently of the main aspects of decision-making*, and therefore yield *an important finding for the metatheory of systemeering*. The results are applied to the systemeering model presented by Kerola and Järvinen (KE/JÄ-75) by formalizing the main aspects (slightly modified) on which the model is based.

2 MAIN CONCEPTS USED IN THE STUDY

In the following a list is given including

1. Definitions of some of the fundamental concepts of system theory and cybernetics
2. Formal definitions of concepts concerned with goal-seeking systems, decision-making or systemeering

Main Concepts of System Theory and Cybernetics

INTERACTION OF OBJECTS, SYSTEM, SYSTEM MODEL

DEFINITION: If in the value sets of some properties of two different objects there are such subsets that in some finite interval of time the properties cannot have values simultaneously in those subsets, we say that the objects are in an *interaction relation between each other*

DEFINITION: A *class object* S containing (only) physical objects is a physical system if the following two conditions hold:

1. Every object in S is in an interaction relation with at least one other object in S (if there are such in S) and every non-empty subset A of objects in S, A $\neq$ S, contains at least one object that is in an interaction relation with some object in S-A.

2. Each object in S satisfies these conditions (1) and (2)

By a *(system) model* we mean a construct of concepts referring to a physical system, its properties or its components and their properties and relationships.

BEHAVIOUR AND ACTION OF A SYSTEM

The word *behaviour* is used in the following four meanings:

External potential bahaviour is the set of all possible courses of action which may be selected by a system (seen from the system environment)

External real behaviour is the particular course of action chosen by a system (seen from the environment)

External realized behaviour is the realization of external real behaviour

Internal behaviour is the process for the selection of the external real behaviour

By the *action of a system* we mean the transition of the values of the properties in the course of time.

GOAL, GOAL-SEEKING, PURPOSEFUL

By a *goal* we mean an identified state of affairs, the realization of which is pursued by a conscious system. A system is *goal-seeking* if it has an ability to state goals and to strive for them.

A state of affairs does not necessarily mean a momentary state of affairs, but generally an identification of features in the development of a system and/or its environment. The word goal, however, is used for a final state regarded as a desirable consequence of the way of operation. When we want to refer to more general features of operation, we prefer the term *purpose*. Similarly, we speak of a *purposeful system* when we want to emphasize the ability of a system to identify such eventual purpose that do not have the nature of means or intermediary goals.

COORDINATION

If a goal-seeking system is composed of several goal-seeking units, we mean by *coordination* the action of a component, the *coordination unit*, that aims at a harmonious cooperation of components.

MAIN POINTS OF VIEW IN DECISION-MAKING

When considering a goal-seeking system composed of goal-seeking units, a particular unit can be examined relative to its external and internal properties. Two of our main points of view, namely the *pragmatic* and *behavioural (I/O aspect)* concern (mainly) external features, while the other two, the *structural* and *operative*, involve internal features.

The *pragmatic study* of a system concerns the effects the way of operation of a system has on the intresents and the external and internal possibilities for operation. One tries to answer the questions:

> Is the system useful for its intresents?
> Why does it exist?
> What kind of real behaviour on the part of the system is favourable for the environment.

In the *behavioural study* of a system one specifies the potential behaviour (I/O relation), the potential ways of operation and their realizability and external consequences. One tries to answer the questions:

> How should the system behave (in the sense of external behaviour) in order to achieve its goals?
> What are the consequences of different ways of operation (relative to the external aspects)?

In the *structural study* of a system one specifies the potential alternatives of the structure and their realizability. In other words:

> How the structure of a system should be developed in order to make the way of operation realizing the stated objective possible?

The *operative study* of a system means the specification of the relationships holding between the external and internal properties of a system. One specifies the potential alternatives for operation, their internal consequences and realizability, and seeks for an answer to the questions:

> How is the stated external behaviour realized by the internal operation?
> How do different ways of operation affect the internal development?

DECISION-MAKING, PLANNING

Decision-making is the process on the mental level that aims at the selection of activities to be performed at each point of time. The word *planning* is used for the systematic construction of alternatives for this selection. One often talks about planning as a process concerning a certain object, in which case it includes decision-making, ie. with planning and selection concerning that object as its subprocesses.

Some Formal Definitions

GOAL-SEEKING SYSTEM

DEFINITON: A goal-seeking system S is

1. a *physical state transition function* f: $X \times K \rightarrow F$, where

X = the set of input-trajectories
K = the set of knowledge state trajectories
F = the set of physical state trajectories

2. a *decision-making relation* $D \subset F \times K$ that determines the potential for decision-making. D can also be presented as a set-valued *decision-making function* k: $F_t \times K_t \rightarrow \mathfrak{P}(K_{(t,t_f)})$ where
   $\mathfrak{P}(a)$ = the set of all subsets of a

3. *an output-function* y: $X \times F \rightarrow Y$, where
   Y = the set of output trajectories

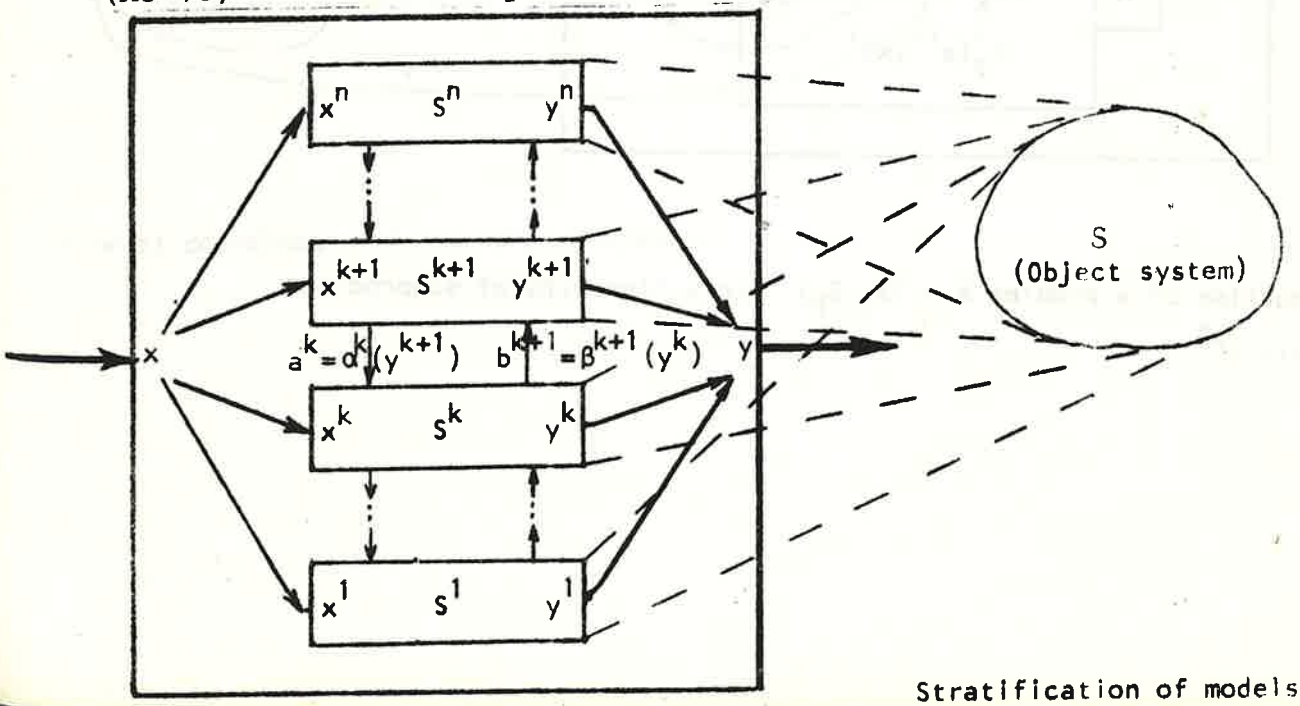
where the decision-making function is not constant with respect to some goals specification included in the knowledge k_t of a system at a given time t.

FORMS OF SYSTEM (MODEL) HIERARCHIES

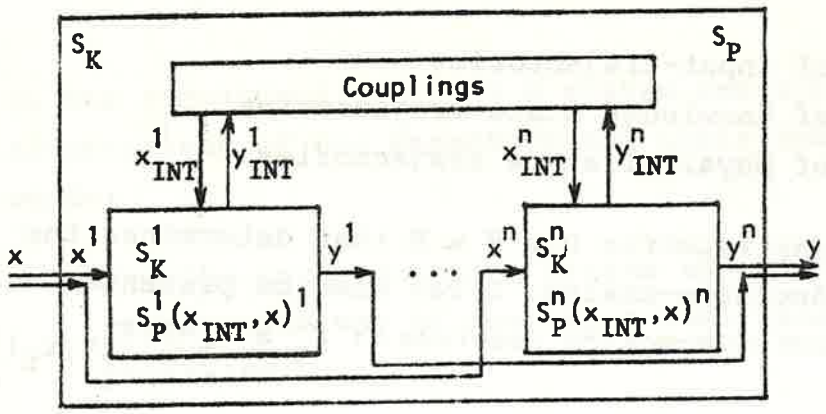
The formal concept of

1. *Stratification (of models)*
2. *Problem hierarchy*
3. *Component hierarchy*
4. *Coordination hierarchy*

are merely illustrated here in the following schematic form. (MU-78) includes the precise definitions.



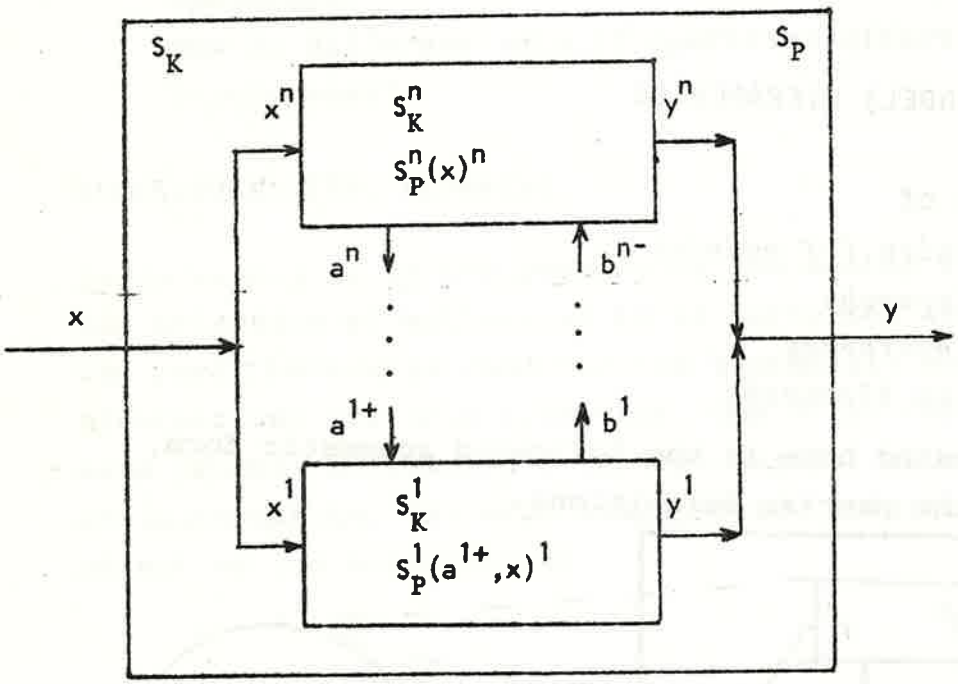
Stratification of models



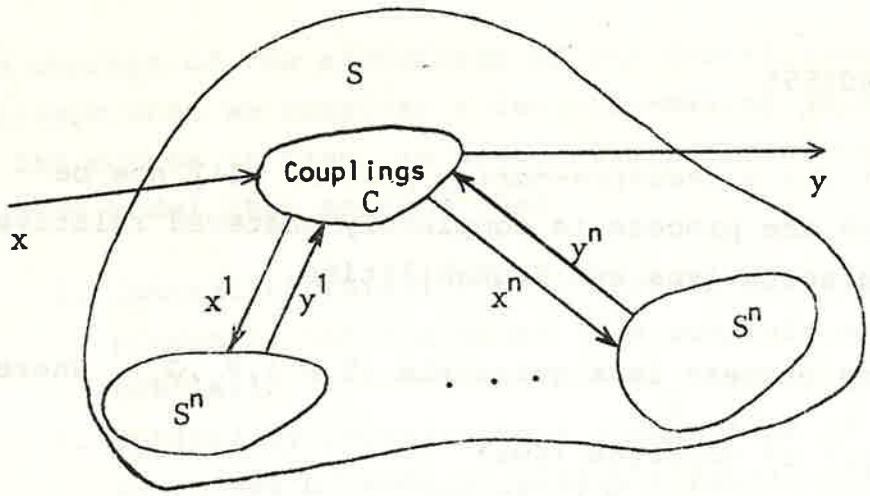
$S^{(i)}_K$ = Model of a solution process

$S^{(i)}_P$ = Solution condition

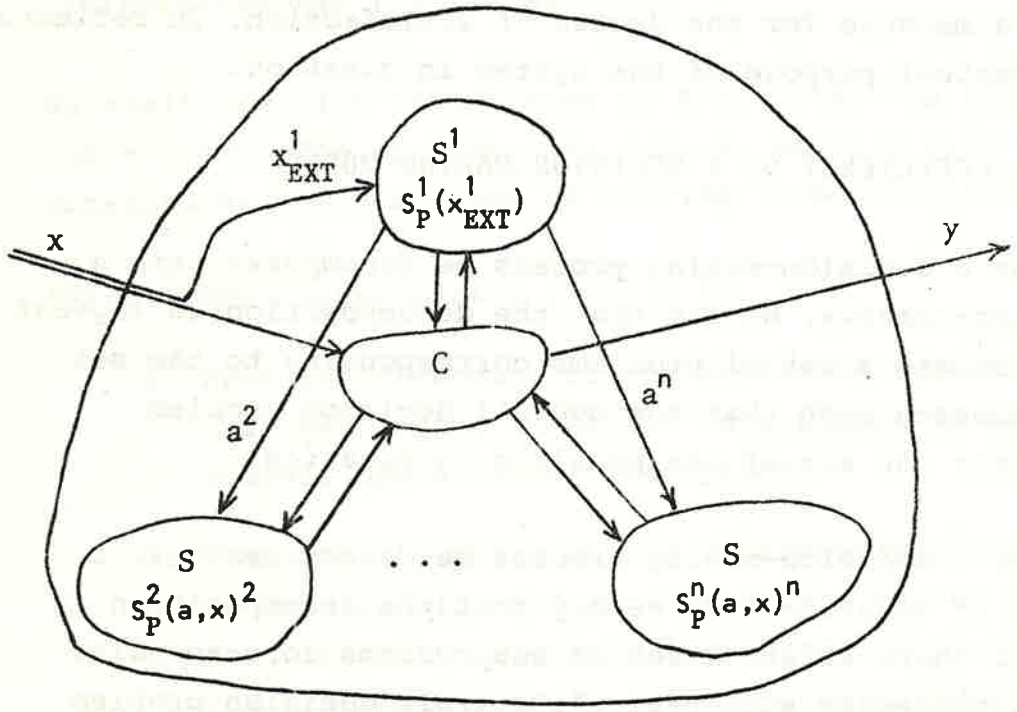
Reduction of a problem $P = (S_K, S_P)$ to a set of subproblems $P^1,\ldots,P^n$



Reduction of a problem $P = (S_K, S_P)$ to a hierarchy of subproblems $P^1,\ldots,P^n$



2-level component hierarchy of systems



2-level coordination hierarchy of systems

DECISION-MAKING PROCESS

A formal metamodel for a decision-making process will now be presented, in which the process is completely mastered relative to the potential alternatives and probabilities.

The decision-making process is a quintuple (T,K,Δ,P_k,G_T), where

1. T is the *set of decision times*
2. K is a *set of states of knowledge*
3. Δ is a set-valued function that determines the potential decision alternatives
4. P_k is a probability function determining statistically the transition of the state of knowledge
5. G_T is a measure for the degree of satisfaction. It reflects the eventual purpose of the system in question.

LOGICALITY AND EFFICIENCY OF A DECISION-MAKING MODEL

DEFINITION: Let a decision-making process be *decomposed into a set of subprocesses*. We say that the decomposition is *logical* if there exists a set of problems corresponding to the set of subprocesses such that the overall decision problem *reduces into the set of the problems in question*

DEFINITION: Let a decision-making process be *decomposed into a hierarchy of subprocesses*. We say that the decomposition is *logical* if there exists a set of subproblems corresponding to the subprocesses such that the overall decision problem *reduces into a hierarchy of the problems in question*

These definitions imply that the *verification of logicality* is relevant when one considers the decision-making process as a *problem-solving process*.

The concept of the *efficiency of the decision-making model* is relevant when we consider a decision-making process in relation to its course in time, as a *learning process*. The efficiency of a given model then means either

1. Optimality relative to a certain class of problems. This presumes that the model is a complete program, not a heuristic
2. Optimality relative to a given class of problems and a given set of models (possibly heuristics)

or

3. The given model is optimal/satisfactory for the "largest" or "most important" class of problems.

SYSTEMEERING AND ITS SUBPROCESSES

We shall now give definitions of a *systemeering problem* and *its subproblems*. The *systemeering process* and *its subprocesses* are obtained as the processes of solving these problems.

DEFINITION: A *systemeering problem* is

1. From the viewpoint of the utilizing system, the notion that developing a certain data system is useful for the overall operation of the system

2. From the viewpoint of the systemeering project, a task of constructing and realize a data system model that satisfies the pragmatic specification of the data system. The pragmatic specification becomes gradually more specific as the process goes on. The task must be completed within the limits of the resource specifications, which also become more specific in the course of time.

A *solution* to a systemeering problem is

1. From the viewpoint of the utilizing system, the notion that it is not profitable to allocate resources to the development on the data system in question

2. From the viewpoint of the systemeering process
   - for each pragmatic specification an estimate of realizability and the resources needed in systemeering
   - for the final pragmatic specification the set (or a subset) of the data system models that satisfy the specification and are realizable within the limits of the resources specified and, if the set is not empty, a realization of one of the models

DEFINITION: A *problem of pragmatic systemeering* is a task of obtaining pragmatic knowledge concerning the data system in question. A solution will contain

1. models that represent the effects of different courses of action in the data system

2. a specification of the modes of operation in terms of external aspects and estimates of the resources necessary in their systemeering

3. selected criteria obtained on the basis of (1) and (2)

DEFINITION: A *problem of I/O systemeering* is a task of obtaining knowledge concerning the external behaviour of the data system. A solution will contain

1. partitions of the I/O specifications produced by the pragmatic systemeering

2. estimates of realizability and the resources necessary in the systemeering of the subalternatives

DEFINITION: A *problem of structural systemeering* is a task of obtaining structural knowledge concerning the data system. One will obtain solutions in terms of

1. alternative structure specifications that potentially correspond to the I/O specification produced by the I/O systemeering

2. estimates of the resources necessary for the construction of the alternative structures

DEFINITION: A *problem of operative systemeering* is a task of obtaining operative knowledge concerning the data system. A solution will contain

1. operative alternatives corresponding to the I/O and structure specifications
2. estimates of the internal consequences of each mode of operation
3. estimates of the realizability of each mode of operation

REFERENCES

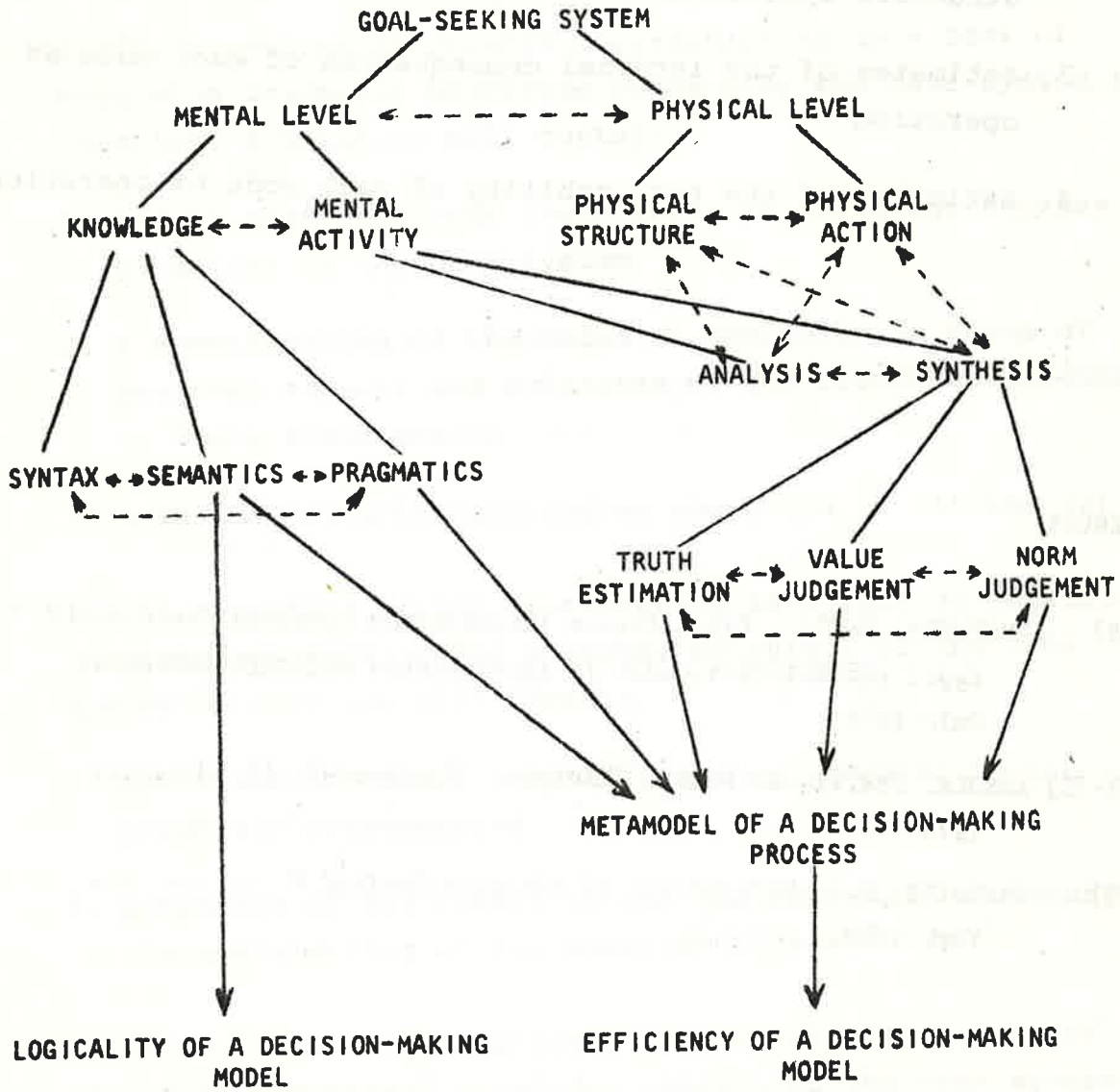
[MU-78] MUSTONEN, SEPPO: *Tavoitteisen järjestelmän kyberneettinen analyysi päätäntäteorioiden ja systemoinnin metatutkimuksessa*, Oulu 1978

[KE/JÄ-75] KEROLA, PENTTI and PERTTI JÄRVINEN: *Systemointi II*, Helsinki 1975

[YA-69] YAKOWITZ, S.: *Mathematics of Adaptive Control Processes* New York 1969

APPENDIX

The concept structure of the study is illustrated here by a graph in which the lines (——) denote analyses, the arrows (——→) predecessor relations and the broken arrows (- - →) causal relations between the phenomena or represented by the concepts.



PROPERTIES AND PROBLEMS OF A GENERAL-PURPOSE DATA SYSTEM

MARKKU I. NURMINEN
The Turku School of Economics, Finland

A general-purpose data system turns out to be closely related to data-bases. In both, the definition and using are separated from each other. The systemeering of such systems is considered, with emphasis on pragmatic and semantic points of view. Further, it is noted that the description of processing must not be ignored even in general-purpose data systems.

Key words: General-purpose data system, Data-base philosophy, Pragmatic and Semantic points of view.

1. The Systemeering Process of a Special-Purpose Data System

In this paper, we distinguish between three different points of view in systemeering, the pragmatic, the semantic and the constructive [1]. The entire process of systemeering may accordingly be divided into three main phases. In the pragmatic phase the emphasis is on the use of data and their effect on the using organization. The central problem is why each particular datum is needed. The main objective of the semantic phase is to define the contents of the system in an operational way. The definition consists of the detailed description of the data processed and produced, their mutual relations, and the processing rules. The description is "logical", i.e. it does not depend on the future physical representation of data or the means of processing them; these are to be determined later, in the constructive phase. The pragmatic and semantic phases can be subsumed under the common term "logical analysis of the system". The constructive phase establishes the corresponding synthesis; the result of the analysis, the logical description, is here mapped onto the real physical world. This mapping results in a feasible model of the data system.

We say that a data system is a special-purpose one, if its output data, i.e. the data needs of the users, are exactly defined at the moment of systemeering.[1] Otherwise we call it a (more or less) general-purpose data system. Two extreme alternatives may be distinguished in the systemeering of a special-purpose system: the data needs may be finally defined either before the semantic phase or after it. If the final choice is postponed until the analysis is completed, an unusually extensive set of potential

data may be necessary in the analysis. The final determination of the data needs cannot, however, be postponed to the synthesis phase, as long as we restrict our study to special-purpose systems [3].

A special-purpose system is clearly characterized by purposefulness. The definition of the data needs uniquely defines the output data of the system, the production of which may be stated as the purpose of the system. The storage of data thus has no autonomous role; data must be stored in order to be able to produce the data needed as the output of the system.

2. The Idea of a General-Purpose Data System.

The computer itself is by nature a general-purpose device. In the main storage of a single computer several different programs may be stored and executed each in turn.[2] Most auxiliary storage devices have a similar characteristic. Computer programs may likewise be considered to have a general-purpose property: a single program is capable of processing a great number of different transactions. There is, however, a certain difference between these two cases. Within the limits of a given program we can change only the values of data terms, whereas the computer itself can be programmed to perform genuinely different tasks. Therefore we do not say that a computer program has a general-purpose property. This is not to say that a program or a set of programs can never have a general purpose, at least to some extent. In many library programs, for example, the actual functioning of the program may be determined at the time of execution by specifying the values of a few parameters. Here

several special-purpose programs have been combined into one general-purpose program. The domain of the possible values of the control parameters defines the class of the possible tasks specific to this program, also called the domain of applicability.

A data system has a similar special-purpose property to that of a computer program. The values of the processed or stored data terms vary, but the structure of the system does not. The generality of a data system, however, may be increased in the same way as we did with computer programs. The data system then becomes an instrument which is capable of performing several different tasks of data production. It may be compared with a general-purpose tool, e.g. a saw, which may be used for many different sawing tasks. The applicability of the saw depends on its properties and on the character of the task. The properties of a particular saw define a class of sawing tasks, i.e. the domain of applicability. The designer of a saw is familiar with this domain of applicability and its necessity.

Each general-purpose instrument has its own domain of applicability. The domain might perhaps be defined by enumerating its elements. The number of potential tasks is often very great, the greater the more general the purpose of the instrument is. It is therefore not always practical or even possible to define the elements of the domain; it may be defined implicitly by specifying the properties of the instrument in question. This makes it possible to decide in any given case whether it belongs to the domain of applicability or not.

Are we justified in introducing the concept "domain of applicability" at all, if we cannot always even define it explicitly? In my

opinion we are, since it is this which gives the instrument, li the special-purpose tool, its purposefulness. An instrument is no instrument unless it is used to perform at least one of the tasks included in the domain of applicability. We do not even know that an object is an instrument if we do not have at least an implicit comprehension of its domain of applicability. It does not seem very reasonable to acquire an instrument which is not known to be an instrument.

The reason for the designing of a general-purpose data system m be discovered in its definition. The exact definition of data needs in the course of systemeering has recently become more an more difficult. The users of data probably have an implicit understanding concerning all potential data which may be necessary. This domain of applicability is often great also due to t fact that single items of data may be combined in many differen ways. Here again it seems practical to define the internal properties of the instrument instead of defining the domain of applicability explicitly. Since the abiguity in output-data need concerns the logical content of data, the internal properties of a general-purpose data system must be accordingly expressed in terms of data contents. Thus, the data stored inside the system have to be determined generally enough to guarantee that the system will be able to produce any output data included in the domain of applicability. This seems to be the idea behind the data-base approach, too. The general-purpose character of the data-base justifies the use in the following of some terms derived from data-base techniques in order to illustrate the problems of systemeering in general-purpose systems.

3. On Systemeering of a General-Purpose Data System

As stated above, the systemeering of a general-purpose data system cannot be initiated by specifying its output data in detail. Rather, the point of departure is the specification of the data stored inside the system. It must be pointed out, however, that this approach likewise cannot be based solely on data and their properties, without any pragmatic or semantic analysis. Even if the pragmatic and semantic analyses do not determine the output data directly, these points of view cannot be totally ignored.

The general-purpose approach logically divides systemeering problems into two categories, those concerned with the definition of the structure and data contents of the system and those concerned with the use of the system.[3] Each of these requires both pragmatic and semantic analyses.

31. Definition of the Structure of the Data-Base

The structural description of a data-base (a general-purpose data system) must contain the specification of at least the following:

what data are stored,
what relations are described,
what terms are used as a key.

None of these specifications can be obtained without a pragmatic and semantic analysis. This has the consequence that the user must participate closely in this specification, because he is actually responsible for its correctness and applicability. The

domain of applicability of the data-base -- even if implicitly defined -- constitutes the ultimate basis for pragmatic justification and semantic definition. Obviously, the participation of the user gives a strong argument for stating the definition in logical rather than physical terms.

Without pragmatic criteria we are powerless to exclude any data of a general-purpose character, e.g. the UN Statistical Yearbook or an encyclopedia. Similarly, the selection of adequate relation relations and keys is a pragmatic problem. Even an implicit comprehension of the potential output data will certainly help considerably. If we happen to know the message type to be formed from the data, it is easy to find the key(s) of the record in question.

The semantic point of view should provide the definition of the data stored. It is practical to construct a globally unambiguous data dictionary for the entire system, in which every data term is given a unique name. This is not sufficient, however, since the system cannot be implemented if the data terms are not de defined operationally. The associative relations do of course establish a structure, but this structure acquires an operational character only if it is supplemented by precedence relations between data terms, with associated processing rules. If the processing rules are not available to the user, he has to search for data by means of data names alone; thus he may obtain and use inadequate data without knowing it.

There is another reason for including the processing rules in the description of the system. Part of the data terms can be excluded from the data-base, if the processing rules and precedence

relations are properly documented. If the precedent data terms are stored, the succedent data terms may be produced whenever desired. This is not to say that such exclusion should always take place. It is, however, important to realize the trade-off situation between storing and processing, which provides an often ignored factor in the aim of controlling redundancy.

32. Use of the Data-Base

If the data needs are not defined exactly during the analysis phase, it must be done later. This definition takes place through a query to the system, with its pragmatic and semantic points of view.

The pragmatic point of view is often straightforward; the data need simply appears. It remains to select the particular subset of the entire domain of applicability now desired. As usual, the pragmatic question why is here closely associated with the semantic question what.

The semantic point of view is now more complicated than the pragmatic one. The query must produce exactly the data desired. This assumes consistency between the query (DML) and the data-base description (DDL). Here the description of the processing rules becomes crucial; if it is not sufficiently detailed, the user may be misled by the incomplete data dictionary. This is likely to occur if the definer and the user are not the same person.

In fact, the processing function of the data-base system may be divided into three parts:

preprocessing, before storage in the data-base,

intermediate processing, both input and output data being stored in the data-base,

postprocessing, generating the output of the entire system.

Preprocessing and intermediate processing should be described together with the data stored in the data-base. Intermediate processing creates the problem of controlled redundancy mentioned above. It is also obvious that the rules of post-processing must be properly defined by the user; this is a pragmatic and semantic task.

4. Summary

The pragmatic and semantic analyses of a special-purpose data system yield the exact definition of the data which it should produce. A general-purpose data system must be able to produce any part of its potential output data set, its domain of applicability. The systemeering of a general-purpose data system is split into two parts, the definition of the data-base and its use. The importance of including the pragmatic and semantic points of view in both parts is one of the main points of this paper.

The fundamental dualism of data systems manifests itself in storage (static) and processing (dynamic). A data system is no system if only storage is described. In a description of a data system, both aspects should be included. The proper description

of processing rules is also necessary for a data-base in order to achieve operational definition of data and controlled redundancy.

The introduction of the concept "domain of applicability" makes sense, even if it cannot be defined explicitly, and the purposefulness of a data-base depends on its existence. In the course of systemeering the domain of applicability constitutes a referent for pragmatic justification and semantic definition. We obviously cannot deny its importance. We therefore suggest that the significance of the domain of applicability should be realized and taken into account more than it has been so far; it might be possible to develop descriptive techniques enabling us to refer to the domain in a more precise and deliberate manner.

GRAMMATICAL DATA BASE MODEL, I DESCRIPTION OF REALITY BY EXPRESSIONS OF A FORMAL LANGUAGE

Eero Peltola, Olavi Maanavilja and Harri Laine
University of Helsinki, Department of Computer Science
Töölönkatu 11, SF-00100 Helsinki 10, Finland

ABSTRACT

This paper describes the basic features of a grammatical data base model at the logical level. The basic assumption in the model is that a logical data base consists of sentences of a formal language. The model also includes the temporal dimension.

Key words and phrases: data base, grammatical data base model, temporal dimension, data model

CR Categories: 4.33

1. INTRODUCTION

This paper is the first in the series of the following three papers:

I Description of Reality by Expressions of a Formal Language

II Fact Expression Language

III Logical Data Manipulation Language

These papers describe our grammatical data base model at the logical (conceptual) level. The theoretical data base models presented in the literature are mostly based on mathematics, especially on set theory. The relational model developed by Codd [1] relies on n-ary relations. The binary association model given by Senko [2] is based on binary relations. Our model is based on the theory of formal languages. Some models, such as those by Bubenko [3], Falkenberg [4], Nijssen [5] and Schmid and Swenson [6], have features in common with our model.

That part of the grammatical data base model presented here considers the data base at the logical (conceptual) level. It is obvious that if we want to implement a data base system based on it we have to extend the model to include other levels of description, e.g. the physical (internal) level. These aspects of implementation will not be discussed here, but will be treated in future papers.

In section 2 we present some general requirements that we have set for our data base model. We have used these requirements as guidelines in designing the model. In section 3 we describe the basic features of the model. The inclusion of the temporal dimension in the model is discussed in section 4. Section 5 contains a summary and conclusions.

2. GENERAL REQUIREMENTS FOR THE MODEL

A given data base describes a specified part of the (perceived) real world. This restricted part of the real world defined by the data base is called a universe of discourse. In order to be able to describe the history of the real world the data base model has to include a temporal dimension. If only the present state of the real world is described, time may be omitted and only the data concerning the prevailing state are retained.

We have set some general requirements for a logical data base. Such a base should be

- natural
- non-redundant
- logically sound
- simple

The word natural is here used to mean that the logical data base should be as direct an image of the real world as possible and thus easily comprehensible by a user, i.e. the data base should describe the real world as the user sees it and not as it is suitable for computer representation. A data base ought to be a non-redundant picture of the real world. This means that at the logical level the same features of the real world are described only once and that those features which can be derived from other features are not described at all. At the physical level this requirement of non-redundancy may, however, be somewhat relaxed.

The data base is logically sound if the similar features of the real world are always described in the same way in a logical data base. If the mode of description is changed in the course of time, the data base should be restructured, from which point it is regarded as a different logical data base. The simplicity of a logical data base means that the logical data base is built from small constituents and the structure of these constituents is simple.

This implies that the structure of the language used to describe the real world at the logical level should be simple. It does not, however, mean that there may not be more sophisticated languages used as interfaces between a user and a logical data base and between a logical data base and a computer.

3. BASIC FEATURES OF THE MODEL

The model has three realms: a real world, a mental universe and a language. Our aim is to form a method for representing the interesting part of the real world (the universe of discourse) by using the language. We call this linguistic representation of the real world a grammatical data base. The mental universe consists of mental images - concepts - in the minds of men. However, we do not regard this realm as equal to the others; rather, we say that it is a frame of reference for interpretation. The mental images connected with the words and sentences of the language form the designations of the words and sentences. It should be pointed out that we regard concepts as mental images, i.e. as something that exists only in our minds. In this sense the words used to stand for concepts are not themselves concepts. Thus if we describe the real world at the conceptual level, this description actually exists only in our minds. A description made using the words standing for the concepts is a different description and is at the language level. We prefer to call this level a logical level instead of a conceptual level as it is frequently called in the literature (cf. a conceptual schema).

The three realms are closely related to the concepts "extensions," "language expressions" and "intensions" used by Carnap [7,8]. Extensions are in the real world, intensions in the mental universe and language expressions are just expressions of the language.

The language consists of discrete elements, viz. words. Thus language can only be used to describe a discrete object at a discrete instant. We therefore assume that a grammatical data base describes the discrete real world at given instants of time, i.e. at given time points. These time points are assumed to be given so that if two linguistic images are identical at two consecutive time points, the part of the real world described by these images does not change

between these time points. In this section we consider the description of the real world at one given time point and therefore exclude the temporal dimension from the description.

The mental universe is used to make the real world discrete for representation. We assume that the mental images or concepts of each person who somehow uses the same grammatical data base are identical, at least with respect to the mental images associated with the data base and the corresponding part of the real world. The users thus have a common mental universe that they use as a frame of reference when interpreting the contents of the data base or when describing the state of the real world by means of the language. Because of the common mental universe when observing both the real world and its corresponding linguistic image they all observe the same errors in the description or they all observe that the description is correct. We shall return to the uniformity of the mental universe in the last section.

In the mental universe there are mental images or concepts. Certain concepts are associated with the words of the language. These mental associations are unique in the sense that one word of the language is associated with one and only one concept in the mental universe. Furthermore, there are also mental associations between the concepts and the objects of the real world, but there may be more than one object in the real world associated with the same concept. The three realms with objects and associations are illustrated in fig. 1.

We may replace these two related sets of mental associations by one set, mentally associating the objects of the real world directly with the words of the language, and say that the concepts form a universe of interpretation. This is illustrated in fig. 2. The mental associations between the words of the language and the objects of the real world state what the references (extensions) of the words are.

gures 1
and 2

Our model is based on the important assumption that a grammatical data base is formed for some known pragmatic purpose. This assumption has the following significant consequences: on the basis of this pragmatic purpose it is possible (1) to extract the universe of discourse from the whole real world; (2) to distinguish the concepts associated with it from all the concepts of the mental universe; (3) to choose suitable words of the language and associate them with these concepts. In practice this may not be a simple matter and there may even be several possible ways of doing it. We are, however, here describing the model and therefore pass over these aspects related to information analysis and hope to be able to discuss them at some other time.

At this point we should emphasize that our model relies heavily on four basic metaconcepts or concepts used to describe the model, namely on the concepts: "object," "class," "association" and "role." According to the model there are objects in all of the three realms and each object belongs to at least one class. Objects within the same realm or different realms may be associated, and in an association each object has a fixed role. We are, however, primarily interested in the objects of the real world, the associations formed by them and their roles in these associations, as well as how these can be described using the language. An association between the objects of the real world appears in the structure of the real world. Words are objects of the language that are used for representing the objects of the real world and their roles. An association of the real world is represented by a sentence of the language, in which the structure of the sentence reveals the position of each word in the sentence. Thus a sentence is an association of objects of the language, in which the general role of each word is defined by the grammar of the language. Sentences are formed as the result of describing real world associations. As word asso-

ciations, therefore, they are of secondary importance. The mental associations connecting the objects and associations in the real world to the words and sentences in the language form the frame of reference for interpretation and are therefore also associations of secondary importance when considering the description of the real world using language expressions.

In the next two subsections we describe in more detail what kind of objects there are in the real world, how they are associated with each other and how the objects and their associations are described using the language. In section 3.3 we discuss the semantic constraints aimed at exposing meaningless descriptions.

3.1 Entities and entity relationships

We stated above that the real world to be described consists of discrete objects and possible associations between them. We now assume that not all these objects (concrete or abstract) are of the same value to us. Some of them we regard as independent objects that we call entities, whereas the other objects are of interest to us only in connection with entities. These latter objects we call descriptors. Next we assume that each entity belongs to one and only one entity class. These disjoint entity classes are called basic entity classes.

The word of the language associated with an entity is called an entity name (cf. proper nouns in a natural language). An entity name is a unique entity name if at any time there exists at most one entity associated with it and at any two different time points the same entity is associated with it (cf. Cesar). An entity name that can be associated with more than one entity at the same time and/or with different entities at different times is called a common entity name (cf. Pope). We assume, however, that entities associated with a common entity name at the same time are regarded

as similar. Just as we must identify these similar entities uniquely, we must also give unique entity names for them. When describing a set of (similar) entities associated with a common entity name, one entity and the number of entities are described. Employees could be one example of entities associated with unique entity names, whereas for parts common entity names (e.g. part numbers) are normally used.

In the language there is a unique word associated with all the entities of a basic entity class. This word we call a basic entity type name. A basic entity type name refers to an arbitrary member of the corresponding basic entity class (cf. appellatives in a natural language).

An entity may be associated with other entities. Such an association is called a relationship. In a relationship each entity has a fixed role, one entity may have two or more different roles and different entities may have the same role.

It is assumed that each relationship belongs to one and only one relationship class. These classes are called basic relationship classes. The word (of the language) associated with each member of the same basic relationship class is called a relationship type name. The word associated with a role in an entity association is called a relationship role name or simply a role name. The same role names cannot be associated with two relationships belonging to different basic relationship classes.

An entity is of interest to us as an independent individual, i.e. without being a member in any relationship. We may say that as an individual each entity forms a one-place association or relationship with the role "an existing member of the basic entity class in question." In the model we assume that a relationship may exist only between existing entities. The existence of a relationship therefore presupposes the existence of the one-place relationship of the entities in question.

A relationship is described by a sentence of the language. We define the language in more detail in part II [10] and use here only examples which we hope will be sufficient to explain our ideas.

Example 1

a) PERSON:[PERSON: John].
b) PERSON:[PERSON: Jill].
c) PERSON:[PERSON: Peter].
d) CAR:[CAR: KP1442].
e) COMPANY:[COMPANY: ABC Ltd].
f) COMPANY:[COMPANY: DEF Ltd].
g) COMPANY:[COMPANY: GHI Ltd].
h) PART:[PART:#227].
i) PART:[PART:#230].
j) PART:[PART:#241].
k) MARRIED COUPLE:[HUSBAND: John; WIFE: Jill].
l) PARENTAGE RELATIONSHIP:[FATHER: John; MOTHER: Jill; SON: Peter].
m) OWNERSHIP:[OWNER: John; OWNER: Jill; CAR OWNED: KP1442].
n) ORDER RELATIONSHIP:[ORDER: D1240; SUPPLIER: ABC Ltd; SHIPPER: ABC Ltd; ARTICLE:#227].
o) ORDER RELATIONSHIP:[ORDER: D1245; SUPPLIER: DEF Ltd; SHIPPER: GHI Ltd; ARTICLE:#230; ARTICLE:#241].

In the above example type names are written in capital letters. Sentences (a) - (j) describe one-place relationships, i.e. they state what entities exist. In these sentences a basic entity type name is used both as a relationship type name and as a role name. Sentences (k) - (o) describe five different multiplace relationships belonging to four different basic relationship classes. In sentences describing multiplace relationships specific words are used as relationship type names and as role names.

From the consistency and non-redundancy requirements it follows that the same feature of the real world can be described neither twice or more times nor in different ways. This means that sentence (l) cannot imply that John and Jill are married, otherwise sentence (k) would be redundant.

3.2 Properties of entities and derived entity classes

A descriptor is an object of the real world that is associated with an entity having a certain role. Thus descriptors are of interest only in connection with entities, and not alone (independently). The descriptor also has a fixed role in the association. This role is called an attribute and the association itself a property association. We say that the attribute - descriptor pair defines a property of the entity with which this pair is associated. If the entity in a property association is a member of a one-place relationship (i.e. it belongs to an entity existence definition), the property is called a general property of the entity in question. If the entity in question is a member of some other relationship (i.e. a multiplace relationship) a property associated with it is called a relationship-connected property of the entity.

A word of the language associated with a descriptor is called a descriptor name. We assume that each descriptor belongs to at least one descriptor class. These classes are called descriptor classes. A word associated with each descriptor of a descriptor class is called a descriptor type name.

A word in the language associated with an attribute is called an attribute name. The word associated with the role of an entity in a property association is, as mentioned earlier, a relationship role name (either a basic entity type name or a proper relationship role name depending on the number of places in the relationship).

An entity may have a property only if it exists. Therefore its general properties are described in the same sentence as its existence as an independent entity. The relationship-connected properties of an entity are described in the same sentences as the corresponding relationships. The following example is aimed at illustrating the description of properties of entities as subordinate clauses.

Example 2

a) PERSON:[PERSON: John(HAIR COLOR: black; EYE COLOR: brown)].

b) CAR:[CAR: KP1442(COLOR: red)].

c) COMPANY:[COMPANY: ABC Ltd(HOME CITY: New York)].

d) PART:[PART: #230(COLOR: green; UNIT PRICE: $12.50; QUANTITY ON HAND: 25000)].

e) ORDER RELATIONSHIP:[ORDER: D1245; SUPPLIER: DEF Ltd; SHIPPER: GHI Ltd; ARTICLE: #230(QUANTITY ORDERED: 15000; SPECIAL PRICE: $10.50); ARTICLE: #241(QUANTITY ORDERED: 15)].

In this example "black," "brown," "blue," "red" and "green" are assumed to be the names of descriptors associated with a descriptor type name "color," whereas "New York" is assumed to be the name of a descriptor belonging to the class of descriptors associated with a descriptor type name "CITY." Furthermore, "$12.50" and "$10.50" are descriptor names of descriptors having a descriptor type name "PRICE," and so on. We assume that descriptor type names can be used as attribute names if this is sufficient for interpretation. In the example "HAIR COLOR," "UNIT PRICE" and "QUANTITY ON HAND" are attribute names used to describe general properties, whereas "QUANTITY ORDERED" and "SPECIAL PRICE" are attribute names used to describe relationship-connected properties. The property "QUANTITY ON HAND: 25000" indicates that 25000 similar parts called #227 exist in the real world described.

We assume that a descriptor name is a word of the language that has a complete and unique interpretation in the mental universe. This means that the attribute name - descriptor name pair is sufficient to describe completely the corresponding property of an entity and thus there is no need to describe the property in more detail. Let us consider a descriptor that is a given amount of money and let the descriptor name associated with it be "5 dollars and 15 cents." One might argue that this descriptor name is composed

of two different descriptor names, namely "5 dollars" and "15 cents." The fact is, however, that neither of these alone is associated with the mental image corresponding to the whole amount of money considered. Therefore we say that a descriptor name is a whole. It may have a fine structure depending on the possible rules according to which it has been formed. This fine structure may be utilized in processing, but it is beyond the scope of this paper.

Each entity belongs to a basic entity class. In the model it is also possible to define derived entity classes by using the general properties of the entities in the definition. Every entity of a derived entity class is associated with a derived entity type name. We illustrate the derived entity classes by means of the following example.

Example 3

a) WOMAN is PERSON with SEX: female

b) REDHAIRED WOMAN is WOMAN with HAIR COLOR: red

This example shows that the definition of a derived entity class is based either on a basic entity class or on a derived entity class already defined. The definition of a derived entity class is very informal. To be more specific we may say that for each derived entity class there exists a derivation rule telling what requirements an entity must fulfil if it is to belong to the derived entity class in question.

3.3 Semantic constraints and derived entity classes

In theory we can assume that the sentences describing the real world are correct. In practice we must be prepared for errors. The first step in this is to demand that all allowable words of the

language be given. We shall discuss in part II [10] how this can be accomplished. The second step is to give rules for the construction of sentences. The basic rules are given by the grammar of the language. These rules are enhanced by other rules that we call semantic constraints. They define what kind of sentences are meaningful.

Semantic constraints are dealt with more closely in part II. Here we discuss only briefly their objectives. In principle semantic constraints are used for two purposes: (1) to tell what kind of relationships an entity may belong to and (2) to reveal what kind of properties an entity in a given role may have. Semantic constraints are expressed as constraints concerning the construction of the sentences of the language.

For each relationship type name we indicate what role names are applicable and the maximum and minimum number of entity names that may be associated with a given role name. For example, it may be stated that in a sentence describing a family there must be at least one and at most two different entity names attached to the role name "PARENT" and from zero to twenty different entity names attached to the role name "CHILD." Thus the "degree of the relationship" need not be fixed (cf. n-ary relations). For each role name we also express the (basic or derived) entity type name of entities whose entity names can be associated with the role name. We also state with how many role names the name of an entity of a given basic entity type can at most be associated in a sentence describing a relationship of a given type, and the maximum and minimum number of sentences in which the same entity name may occur. Thus the semantic constraints can be used to express, for example, that in a "MARRIED COUPLE" sentence both the role "HUSBAND" and the role "WIFE" must occur once and only once, that the same person cannot be both "HUSBAND" and "WIFE" (i.e. the same person can occur only once in the sentence) and that the same person may be refer-

enced in only one "MARRIED COUPLE" sentence describing a given instant of time.

For each role name (and therefore for each basic entity type name) the attribute names that can be used in a subordinate clause associated with the role name are stated. In addition, the maximum and minimum number of times each attribute name may occur in the clause is given together with the type of descriptor whose name is associated with the attribute name. Thus the semantic constraints can be used to express, for example, that "COLOR," "UNIT PRICE" and "QUANTITY ON HAND" must be described for each "PART," or that for each ordered part, i.e. for each "ARTICLE," at least "QUANTITY ORDERED" and possibly "SPECIAL PRICE" ought to be described.

4. INCLUSION OF THE TEMPORAL DIMENSION

In the previous section we discussed how to describe the real world by means of a language at a given time point. Now we consider how to describe it in the course of time. We regard time as a discrete parameter in the model. This implies that we have to describe the real world at given points of time, these points being chosen appropriately. The properties of entities and relationships between entities are not, however, the only ones changing with time. The universe of discourse changes, too. Entities in the basic entity classes appear and disappear. New entity classes may be defined as well as new properties and relationships concerning the old or new entities. Descriptor classes may also change. This means that semantic constraints as well as sentences must include time as a parameter. On the other hand, the basic structure of the language and the mental associations connected with its words are assumed to remain unchanged, otherwise we would not be able to say what the sentences were describing.

A unique word of the language is associated with each time point considered. These words are called temporal names (cf. temporal adverbs in a natural language). Time is included in our model by simply adding a temporal name to each sentence; for clarity we associate the temporal name in the sentence with the type name "TIME." The temporal name tells us when the feature of the real world described by the sentence was observed. We must, however, ensure that time is interpreted correctly. We assume that in the model the existence of an association detween objects of the real world presupposes the existence of the objects in question. Let us use here years as temporal names, i.e. as names of time points, and look at the sentence "FAMILY:[MEMBER: John; MEMBER: Jill; MEMBER: Peter] TIME: 1974." The sentence tells us that in the year 1974 John, Jill and Peter formed a family, provided that we have defined the sen-

tence as describing a family relationship. This is done by giving the corresponding semantic constraints and expressing (in the "data dictionary") the meanings of the related words. According to the model, the sentence does not imply that these persons formed a family in the year 1975. This must be expressed by another sentence. If we have to describe a biological relationship between these persons we may do it by means of the sentence 'PARENTAGE RELATIONSHIP:[FATHER: John; MOTHER: Jill; SON: Peter] TIME: 1972.," where we must agree how to interpret the sentence. We may define the sentence as describing the birth of Peter. According to this definition the sentence "PARENTAGE RELATIONSHIP:[FATHER: John; MOTHER: Jill; SON: Peter] TIME: 1973." is semantically incorrect, because we know that Peter was born only once. Thus in the corresponding semantic constraint we must state that any entity name referring to an entity of type "PERSON" may occur in only one sentence when associated with the role name "SON" (or "DAUGHTER").

Let us next consider how time is connected to the semantic constraints. This is done simply by stating at what time points each constraint is applicable. In addition, we state for each entity name and each descriptor name (except the temporal names) at which time points (i.e. concerning which temporal names) they are applicable. A sentence is incorrect if any entity name or descriptor name included in it, or any semantic constraint concerning it, is not applicable.

This conduct has an important consequence: on the basis of the time expressions it is possible to say by means of which temporal names given properties or relationships of a given entity might be described by the sentences in the data base, without looking at those sentences. The applicability of an entity name does not imply the existence of the entity; for example, the entity name KP1442 may be an applicable entity name for a car although no car with

this name exists at present. The existence of an entity at a certain time point is described by means of a sentence describing it as an independent entity. This is analogous to the situation in programming languages. A variable may be declared without being used in any statement, but a variable must be declared before being used in any statement.

At this point we should again emphasize that the aim of the model is to describe the real world at the logical level. At other levels it may be more practical to use time intervals instead of time points. A time interval is the period between two time points. The external (user) and the internal (computer) level might accommodate time intervals.

Finally, we should note that in practice a data base probably only partially incorporates the temporal dimension. The part of the real world to be described is then divided into two subparts. Only the present state of one subpart is described, whereas both the present and past states of the other part are recorded. This may be accomplished by stating in the semantic constraints which sentences must contain the temporal expression explicitly. The sentences without a temporal expression are then interpreted to describe the current state of the real world.

5. CONCLUSIONS

In this paper we have presented the basis of the grammatical data base model: how the real world is described by sentences of a formal language. This description follows the use of a natural language and is therefore quite natural to the user. Models emphasizing the linguistic features of the description have been published earlier (cf. for example [4] and [5]), but our model contains many features not included in those models. A convenient, compact comparison of various data base models is presented by Senko [9] and this could be used if a quick comparison of our model with other models is required. A short presentation of some models dealing with the temporal dimension is given by Bubenko [3].

The grammatical data base model regards the universe of discourse as consisting of two kinds of objects: entities and descriptors. It might be argued that it is not always easy to know whether an object in the real world should be regarded as an entity or as a descriptor. We feel that in many cases it is easy to decide, and that it is always possible to define the object to be an entity. Thus the use of descriptors is not a restriction, but instead helps to make the model more natural. (We think it unnecessary to regard both a color as an attribute of a car and a car as an attribute of a color.)

In the model, entities may have general and relationship-connected properties. In order to make the model simple and to include as much semantics as possible in it, a relationship cannot have a property; however, an entity constituting a relationship may have relationship-connected properties.

The model allows the use of n-ary associations between entities, even to the extent that the number of associated entities may vary within relationships of the same type. It should, however, be emphasized that the relationships are guided by semantic constraints

and these constraints may be used, for example, to make all the relationships binary.

In the model it is assumed that each word of the language used to describe the real world has a unique meaning, i.e. each word is associated with a unique mental image that in turn can be associated with one or more objects in the real world.

These associations contain the meaning of the word. The aim of the model is not to describe the mental images, but rather the real world by means of known mental images. The model does not fix the words of the language. We assume that the words will be given by the different users of the data base in question. For example, the entity and descriptor type names, temporal names, attribute names and other role names, as well as semantic constraints, might be defined by a data base administrator and possibly approved by an enterprise administrator, whereas entity and descriptor names might be given by end-users and possibly approved by the data base administrator.

In section 2 we stated that the mental universe of all the users must be unique, at least with respect to the mental images concerning the logical data base they are using. We may relax this requirement somewhat. Those users who observe the real world and describe it by means of sentences of the language must have in their minds all those mental images they need in their work. They must also know the mental associations between the mental images and the objects of the real world, as well as between the mental images and the words of the language. For those users who only process the logical data base but do not observe the real world it is not necessary to know the associations between the mental images and the objects of the real world. For example, one may calculate Peter's age without knowing him.

The model does not include the use of synonyms. We feel that the requirement of a unified description prevents the use of synonyms

at the logical level. There is, however, no such restriction on the use of synonyms at the external (user) level.

In part II [10] we define a formal language which we have given the name FEL, Fact Expressing Language. The set of sentences of this language describing a desired part of the real world is a logical data base. In part III [11] we define a logical data manipulation language the purpose of which is to process a logical data base. This language forms the core around which more user-oriented (external) updating and query languages are to be built.

A logical data base is virtual in the sense that it does not exist as such in a computer. There is another formal language at the internal level (possibly several, if one wants to define intermediate levels). We call this language FSL, Fact Storage Language. Sentences in FEL are translated into sentences of FSL when stored. The logical data base expressed in FSL is called a physical data base. Correspondingly, there is a physical (data base) manipulation language that is used to carry out the operations defined by statements of the logical manipulation language.

A data base is a data store that is meant to be used. Its use can be divided into two parts: new sentences are added to it and existing sentences are retrieved from it. The former use (updating) presupposes that the sentences to be added must also be expressed at the physical level. It would be desirable to use FEL and FSL for this purpose. In practice, other more expressive languages may be needed. The latter use (querying) requires that the retrieved sentences should be stored for processing. Again we need at least one language for expressing the retrieved FEL sentences in a stored form. This language will probably have to be capable of expressing how new data (sums, counts, etc.) are formed from the retrieved sentences. We plan to consider these external updating and query languages in future papers.

REFERENCES

1. E. F. Codd: A relational model for large shared data banks. *Comm. ACM*, 13(6), 377-387,(1970).
2. M. E. Senko: The DDL in the context of a multilevel structured description: DIAM II with FORAL. *Data Base Description*, B.C.M. Dongue and G. M. Nijssen, eds., North-Holland Publishing Co., Amsterdam, 239-257, (1975).
3. J. A. Bubenko Jr: The temporal dimension in information modeling. Research Report RC 6187, IBM Thomas J Watson Research Center, (1976).
4. E. Falkenberg: Concepts for the coexistence approach to data base management. *Preprints of the International Computing Symposium 1977*, E. Morlet and D. Ribbens, eds., North-Holland Publishing Co., Amsterdam, 39-50, (1977).
5. G. M. Nijssen: Current issues in the conceptual schema concepts. *Architecture and modelling in database management systems*, G. M. Nijssen, ed., North-Holland Publishing Co., Amsterdam, (1977).
6. H. A. Schmid and J. R. Swenson: On the semantics of the relational data model. *The Proc. of the International Conference on Management of Data 1975*, E. F. King, ed., 211-223, (1975).
7. R. Carnap: *The Logical Syntax of Language*, Kegan Paul, Trench, Trubner & Co. Ltd, London, (1937).
8. R. Carnap: *Meaning and Necessity, A Study in Semantics and Modal Logic*, Phoenix Books, Chicago, (1956).
9. M. E. Senko: Conceptual schemas, abstract data structures, enterprise descriptions. *Preprints of the International Computing Symposium 1977*, E. Morlet and D. Ribbens, eds., North-Holland Publishing Co., Amsterdam, 85-102, (1977).
10. O. Maanavilja, H. Laine and E. Peltola: Grammatical data base model, II Fact Expression Language. *this issue.*

11. H. Laine, E. Peltola and O. Maanavilja: Grammatical data base model, III Logical Data Manipulation Language. *this issue*.

(Figure Captions)

Figure 1. An illustration of the three realms of the model with examples of objects and their associations.

Figure 2. An illustration of the two basic realms of the model with examples of their objects and associations between the objects. The third realm, the mental universe, acts as a universe of interpretation.

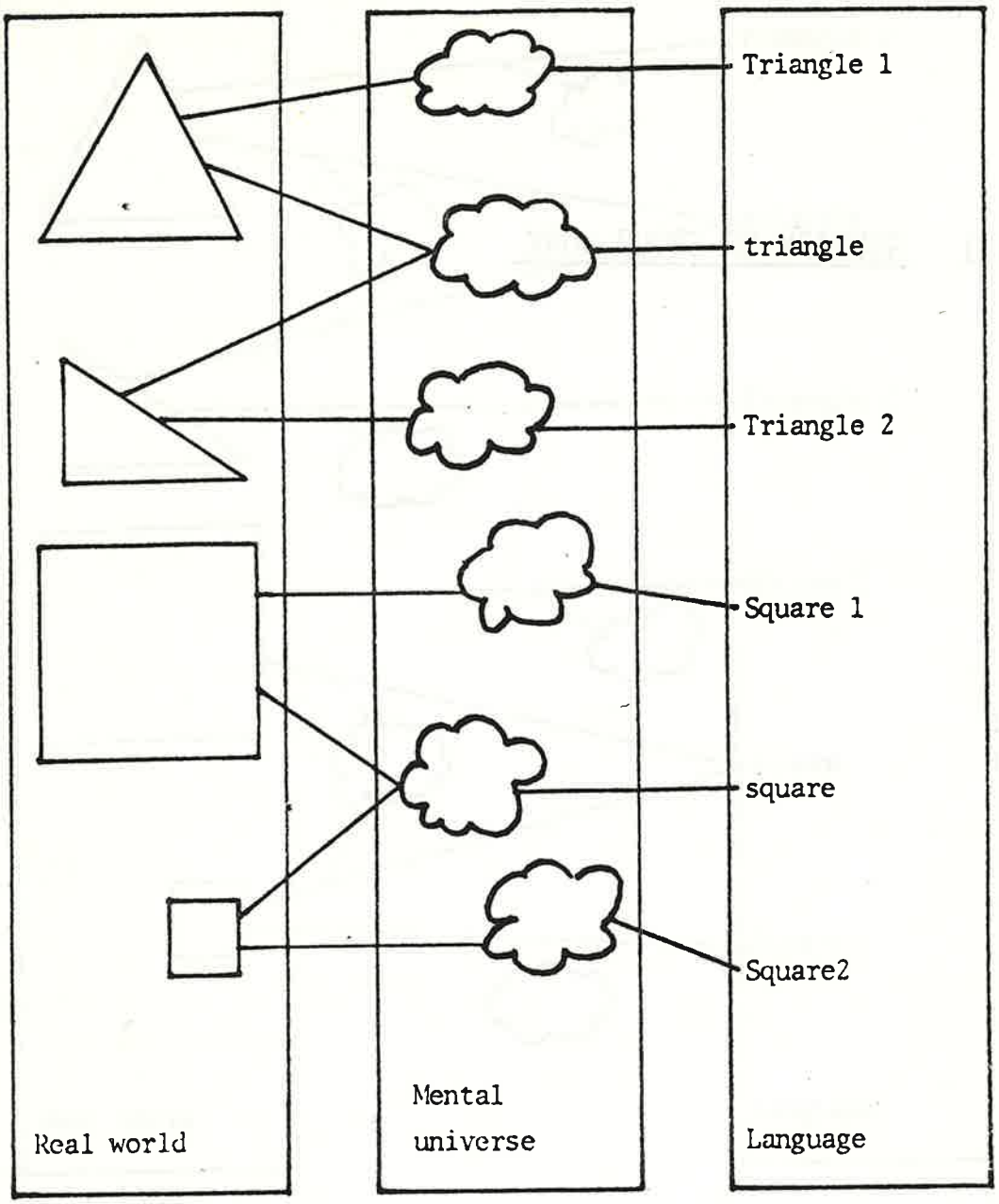


Figure 1. An illustration of the three realms of the model with examples of objects and their associations.

III SUMMARY OF GROUP WORK

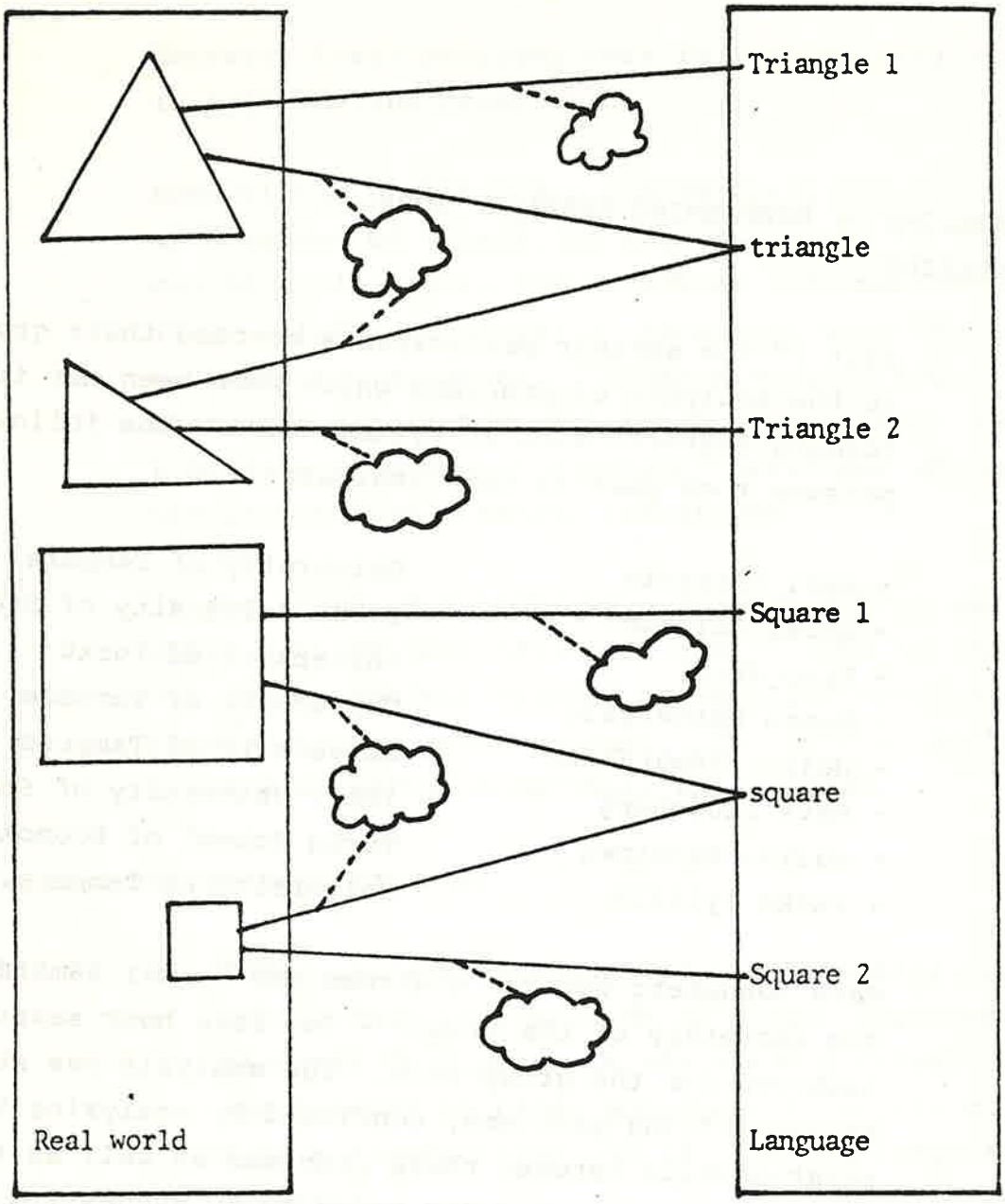


Figure 2. An illustration of the two basic realms of the model with examples of their objects and associations between the objects. The third realm, the mental universe, acts as a universe of interpretation.

G1 ON THE PROBLEMS OF PERFORMING RESEARCH WORK IN SYSTEMEERING

Half of the seminar participants devoted their group work to the analysis of problems which have been met in performing research work in systemeering. The following persons took part in this analysis:

- Kari Eloranta	University of Tampere
- Göran Goldkuhl	ISAC, University of Stockholm
- Timo Järvi	University of Turku
- Hannu Kangassalo	University of Tampere
- Heikki Kämäräinen	University of Tampere
- Mats Lundeberg	ISAC, University of Stockholm
- Markku Nurminen	Turku School of Economics
- Pekka Tyllilä	University of Tampere.

Mats Lundeberg was the chairman and Heikki Kämäräinen the secretary of the group. - Two five hour sessions were reserved for the group work. The analysis was started by identifying problems, continued by analyzing the relationships between these problems as well as the priorities among them and ended up by suggesting solutions to the most severe problems.

G1.1. Problems that plague research work in systemeering

The task of the group was to concentrate on the meta-problems of systemeering research work; the object-problems, which are identified and solved during actual research activity, were left aside. Consequently, most of the identified problems are of general nature and in no way characteristic of only systemeering research work.

However, these problems were felt to be very real also in systemeering research.

Some of the problems are due to the youth of systemeering as a branch of scientific inquiry. Some of the problems can be explained by the nature of systemeering as a science (e.g. natural v. social v. technical science; descriptive v. normative science). It was also understood that not everybody experiences all the problems: e.g. some are typically problems of a junior researcher whereas others are problems of a senior researcher.

In the search for problems a classification was used, which divides them into problems related to research activity, problems related to research results and problems related to research contacts. In addition to these the group also gathered personal experiences.

Problems related to research activity

A1 There is lack of education in performing research work. This is a problem of junior researchers. Also teachers find research education problematic; it is not trivial to raise competent researchers. It was felt that education should cover at least the following issues:

- what is research all about?
- what research philosophies/perspectives (appropriate to our field) are there?
- what research methods/approaches (appropriate to our field) are there?
- how to define concepts as well as construct and test theories?
- how to communicate (e.g. research results) in a research environment?
- what is the identity of our research field?

A2 There is lack of educational background. Many young researchers come to realize that their education has not included studies which are vital for their career. E.g. studies in languages, philosophy, mathematics can be such missing assets.

A3 There is lack of practical experience. This can be a problem of both junior and senior researchers.

A4 It takes time, resources and energy to establish a follow-up network of scientific publications and a network of contacts to other researchers and to the practice. This is a problem of young researchers.

A5 There is lack of time - or rather: it is difficult to determine an order of priority for the multitude of beneficial activities and to gain a proper balance between the most important activities.

A6 There is lack of research funds, research equipment etc.

A7 It is difficult and resource demanding to test ideas. Data processing systems are mostly part of organizations. Consequently idea testing often requires experiments in organizations. It can be difficult to find an organization which is willing to take the risk; the experiment might last for years; the experiment might imply big, costly changes in the organization etc.

A8 There is a gap between research and practice. This gap can be due to the fact that researchers have drifted away from concrete problems to their abstract model world or that researchers are too far ahead practice. In both cases researchers have difficulties in establishing and fostering co-operation with practice as well as in marketing their results to practice.

A9 There is lack of motivation for research work. This can be due to hard experiences, relatively scant compensation etc.

problems related to research results

R1 It is not clear, what proper research results are. Often (at least young) researchers have too high ambitions. Practical research work is often considered as second class research and thus neglected.

R2 Young researchers easily adopt inappropriate analogies from other sciences.

R3 Choosing the length of research paper is difficult. If a short paper is written, it gives rise to misinterpretations; if a long paper is written, nobody reads it.

R4 Practical research work is difficult, because people engaged in systems development resist changes. This is paradoxical, because the duty of systems development people is to change the practice of other people.

R5 Researchers often have to live with poor secretary services.

R6 Especially mathematical background leads to unrealistic expectations of exact results in systemeering research.

R7 If the research project is long, there is a danger that the results are obsolete and out of date, when the project comes to an end. Rapid development of data processing technology and intensive + extensive research efforts make these unpleasant events possible.

R8 Researcher has sometimes grave difficulties in promoting practical applications of his research results.

R9 There is lack of journals where to publish systemeering research results.

R10 There is lack of public, if the research problem is spesific or advanced enough.

Problems related to research contacts

C1 There is no proper research environment, i.e. there is no group of competent and motivated fellow researchers to work with. At least four possible causes were found for this problem:
- too many other duties
- the academic remuneration system (we are remunerated on the basis of our individual accomplishments; not e.g. on the basis of our contributions to team work)
- the academic fear for not being the first one to publish research results
- the fear that colleagues steal ideas.

C2 There is no guidance for a young researcher. Therefore there is a great danger that he makes unnecessary, especially "bad" mistakes i.e. mistakes, from which he does not learn anything.

C3 There is lack of contacts to practice. Contacts are most often of personal nature, not of institutional one.

C4 There is lack of international contacts.

C5 There are communication difficulties between infological (user-oriented) and datalogical (computer-

oriented) researchers.

c6 Co-operation with students in the systemeering field is not without problems, e.g. because students prefer formalized subjects to unformalized ones.

Experiences from research work in systemeering

E1 It is difficult to find applicable results from other scientists. Consequently we tend to define our own concepts and erect our own theory constructions from scratch.

E2 Research work is iterative. It is unrealistic to expect to get the things right in the first draft of the report. Reports are normally rewritten a few times.

E3 It is important to think, to whom we address our paper. This determines to what extent it is necessary to explain the concepts and thought models that are used in the paper.

E4 It is time consuming but necessary to follow, what goes on in our field.

E5 Seminars can be stimulating. Group work can be rewarding.

E6 Good ideas don't often get a warm welcome.

E7 Research project management is difficult.

E8 Closeness to reality is beneficial for hypothesis generation.

E9 Rejection of hypothesis does not mean failure of research project.

E10 Researchers are not always considered as human beings. E.g. a researcher might be given notice of the discontinuance of service in a way that would be completely unacceptable outside academic world.

E11 Performing research work is a 24 hour job.

E12 Research work is stimulating to personal development.

E13 Co-operational research projects in companies tend to get a low priority, when they are free of charge.

G1.2. The most severe problems and suggested solutions to these problems

The next step in the group's work was to carry out a close analysis of the identified problems. The problems were classified in a few ways; the (causal) relationships were analyzed and visualized; priorities were given to the problems. As an outcome of the group work three most severe problems were picked up, the first one of which is the most severe:

A1 There is lack of education in performing research work.

C1 There is no proper research environment.

A3 There is lack of practical experience.

The consequence of these problems is lack of competence in performing research work. The group felt that the competence of systemeering researchers can in most cases be significantly improved. The improved competence would solve some of the problems right away and make it easier

to solve many others; e.g. it is much easier for a truly competent researcher to acquire resources for research than for his less competent colleague.

In the following, solutions are suggested to these three basic problems. Control activities (C) are separated from the activities (A) that are controlled.

Promoting education (ED) in performing research work

C-ED0 Planning of our own education. Whatever our research environment is like, we can always educate ourselves.

C-ED1 Identification of the educational need.

C-ED2 Seminar for education development.

C-ED3 Experimental program on performing research work.

C-ED4 Planning for regular courses.

C-ED5 Identification of relevant literature.

C-ED6 Working groups for establishing research methodology.

A-ED1 Better selection of supporting subjects during undergraduate and graduate studies; courses in the curriculum on performing research work.

A-ED2 Own activity e.g. pilot research projects and study of relevant literature.

A-ED3 Real research assistant positions. Nowadays research assistants are assistants to nobody, i.e. they are independent researchers. During the first years of research work a close relationship to a

senior researcher might be beneficial.

A-ED4 Guided group work in postgraduate studies.

A-ED5 Doctoral seminar. In small countries these could be open to other universities as well.

A-ED6 Departmental or faculty seminar on performing research work. The use of guest speakers would make the arrangements easier.

A-ED7 Nordic summer course on performing research work. Foreign lecturers could be used.

(As to the contents of education, see problem formulation A1.)

Promoting research environment (ENV)

C-ENV1 Identification of desired research environment (individual/university/city/nation/Nordic countries)

C-ENV2 Local development projects for improving research environment.

C-ENV3 Exchange of experiences from the above development projects.

A-ENV0 Own activeness e.g. in making initiatives and in participating in groups, seminars etc.

A-ENV1 Daily informal contact to 1-2 researchers.

A-ENV2 Weekly research seminar e.g. for follow-up of the individual research efforts, for common study of books or for literature survey.

A-ENV3 Development of national contacts between researcher

A-ENV4 Dvelopment of formalized contacts between Nordic researchers.

A-ENV5 Exchange of young researchers between universities.

A-ENV6 Grants for studies abroad.

Promoting practical experience (EXP)

A-EXP1 Summer jobs during education.

A-EXP2 At least one year's practical experience required for researchers.

A-EXP3 Action research projects.

A-EXP4 Empirical research projects.

A-EXP5 Seminars and working groups where practitioners and researchers meet.

A-EXP6 Leaves from university.

A-EXP7 Part-time jobs.

A-EXP8 Long term personal/institutional relationships to organizations.

A-EXP9 Formal contracts between department and organizations.

In the group it was understood that this sort of discussions easily lead nowhere; the awareness of potential problems grows, but the attitudes and the behaviour nevertheless does not change. It was clearly recognized that we must not wait for a remedy from outside ourselves. Consequently a number of commitments to improve our competence as researchers were made during the seminar.

Eds. Kalle Lyytinen
Mikko Klemola

REPORT OF THE WORK GROUP 2 - DEFINITION OF SOME BASIC CONCEPTS

1. INTRODUCTION

Work group 2 worked on two sessions. Tasks of the group were

- to discuss and to define some fundamental concepts essential for systemeering research
- to analyze the nature, aims and content of systemeering
- to use this analysis as a starting point to define the concept of systemeering and argument it.

Members of the group were

Pentti Kerola (chairman 2nd day)
Eero Peltola (chairman 1st day subgroup 1)
Pertti Järvinen (chairman 1st day subgroup 2)
Seppo Mustonen
Pekka Tyllilä
Mikko Klemola
Kalle Lyytinen

On the first day above mentioned topics were discussed in two separate subgroups:

- the purpose of the first subgroup was to characterize the nature systemeering and to derive possible definitions for systemeering concept
- the purpose of the second subgroup was to define some basic concepts of systemeering especially data and information.

On the second day these groups worked together concentrating their efforts solely on the problem of the nature of systemeering.

In the following the results of these work groups will be reported in two sections. In the first section the results and proposed preliminary definitions for some basic concepts will be presented. Comments and discussion on these concepts are given.

Finally some aspects about the problems of concept definition and the dynamics of group work attaining to define these concepts will be mentioned.

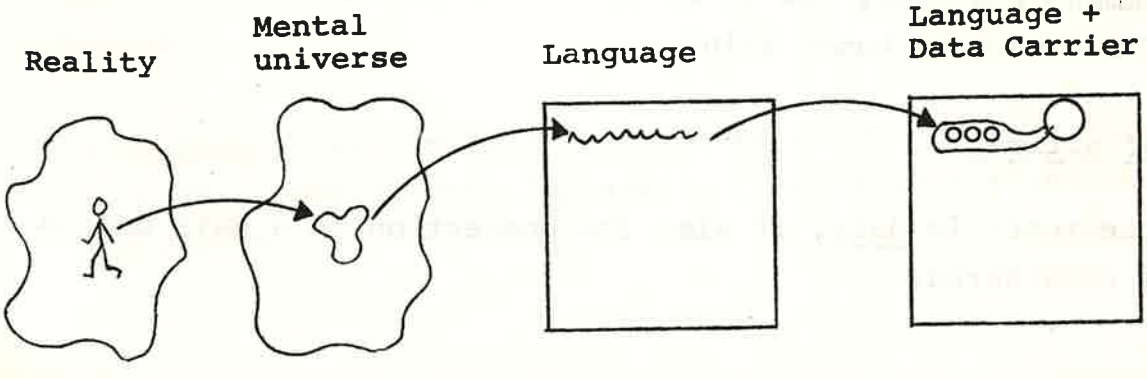
2. DEFINITION OF SOME BASIC CONCEPTS RELEVANT TO SYSTEMEERING RESEARCH

The basic concepts discussed were:

- data
- sentence
- information
- sentence system
- data production system
- sentence production system
- information system
- system
- structure
- model

The work group emhasized the immediate need for developing a preliminary set of some basic concepts. For that reason, in order to get work started the work group laid severe restrictions on the scope of applicability of definitions and used as a starting point the following idealized situation. (see fig. 1)

Figure 1. The frame of reference for definitions



We have in reality (object system, universe of discourse etc.) some things which are potential objects for description in some formal language. The symbols (symbol strings) used in the alphabet of formal language can be attached to some things in reality via mental universe. No attention in this conection is paid to the problems how this connection is established or how we have arrived to the formal language. In this way a connection is established between the observed states of affairs in reality (things having properties and/or taking place in relations to other things) and abstract structures of a formal language. In this frame of reference the work group made the following propositions for definitions of data, sentence, and information.

Definition

A sentence is a linguistic expression describing a state of affairs in real world (a given object system) when using a specified (formal) expression language.

Discussion

During discussion some comments regarding the selection of appropriate term was given. Other suggestions for the term were 'fact', which was almost totally rejected because of its ambiquity. Other suggested terms were 'proposition' and 'statement', but these were also considered to have more ambiquity than the term sentence. One problem was the inclusion of time to the definition. If we regard, that states of affairs exist only in time, because reality changes in time, then the concept of time is implicitly implied in definition. The point is then only, must we define it explicitly. One comment was also, how to state in the definition the receipt of sentence by human being.

Definition

A sentence is data, if also its connection to a data medium is considered.

Discussion

This definition means, that if the physical structures realizing abstract linquistic structures are known and implemented, these implemented physical structures conveying linquistic structures are data.

Definition

A sentence or a set of sentences not known to a user is regarded as information for this user.

Discussion

Some participants of the seminar regarded this as an interpretation by Shannon-Weavers information theory, although this was not the purpose of work group's definition. Some participants also emphasized that newness of the sentences is not sufficient criterium to consider them as information. The pragmatic effects must also be taken into consideration eg. as in Ackoff's communication theory.

Definition

A sentence system is an abstract system defined by the syntax and semantics of the expression language.

or

Definition

A sentence system is an abstract system which is a subset of all possible sentences of the defined expression language.

Definition

A Data Production system is a real system which with the given resources (data processors) implements (realizes) the defined sentence system.

Definition

A sentence production system is a system composed by the end user(s) and the data production system whose end user is a source of trigger and control

Definition

An information system is a sentence production system where the sentences produced are information.

(A priori the aim of sentence production system is to produce sentences which convey information.)

These concepts are quite obvious and need not more explanations.

In addition to the previous definitions which characterize the essence of the phenomenon which is the subject of research in systemeering the work group defined the following concepts which are necessary but not sufficient to analyze and synthetize the phenomena in systemeering research.

Definition

A system is a defined (possibly changing) set of objects and a defined (possibly changing) set of relations between these objects in the course of time.

Discussion

At this point several problems concerning the definition aroude. Those were

- the generality of the definition (extension, applicability)
- must the purpose of the definition be clearly present
- how the adjective "systematic" is shown in the definition
- how can the classes of systems be represented in the definition
- is a thing, object of examination seen as a system or is it in itself a system

Definition

A structure of the system is formed by its relations.

Discussion

If the relations are all stable the system is static. Otherwise it is dynamic i.e the structure of the system changes during time. The cardinality of relations can be infinite. Also some restrictions can be laid upon the objects which can participate in certain relation.

Definition

A system S' is regarded to be a model of a phenomenon S, if a correspondence is specified between the objects and the relations of these objects so that S' can be used in answering questions concerning S.

or

"Object A^t is a model of object A to subject B to the extent that B can be use object A^t to answer questions that concern object A" (Minsky, 1968; cf Kämäräinen's report in these proceedings).

During discussion some participants questioned about the "substance" or realm on which the model is described. Is the realm mental, linquistic, physical etc.? The main conclusion was that models can appear in all realms. It mainly depends on the particular view taken by the observer what is to be considered as a model.

3. DEFINITION OF THE CONCEPT SYSTEMEERING

3.1. First day group

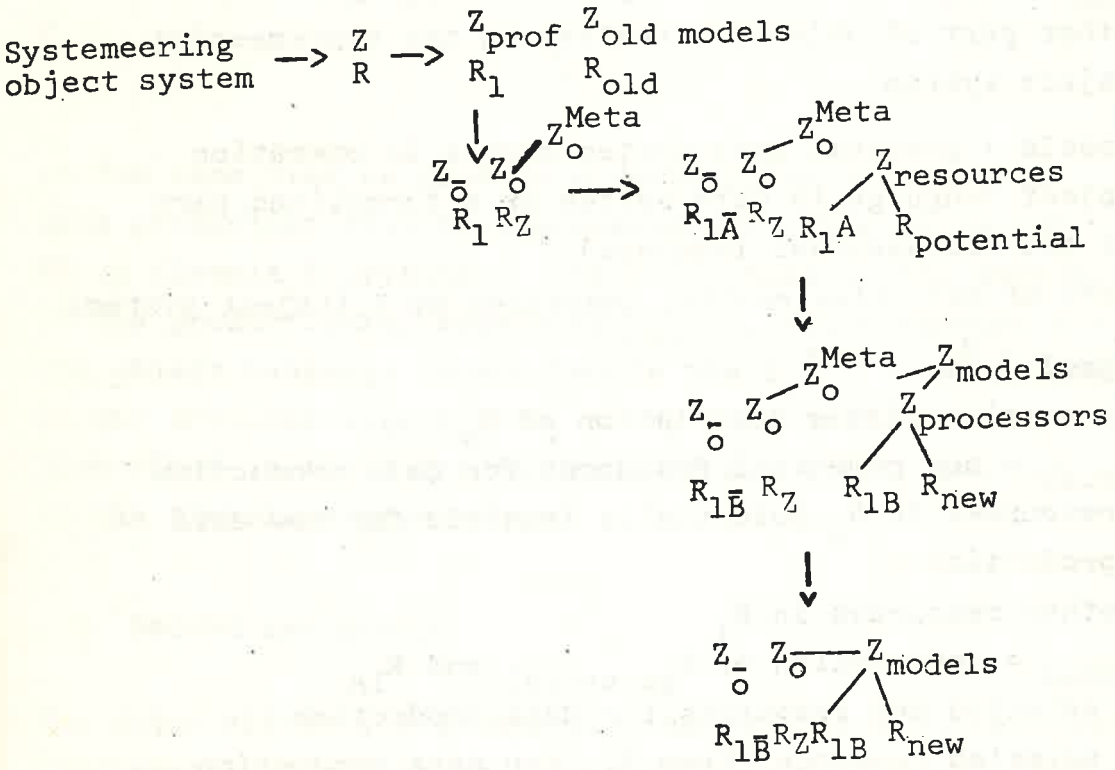
The group strived for specification of phases of systemeering. The result was the schema (see fig. 2) illustrated in the next two pages. Concerning to the used terminology we refer to the preliminary paper of Klemola and Lyytinen (cf pp. of this book).

The meaning of the schema is as follows. In the systemeering object system we can distinguish different kinds of languages (Z) and other parts of systemeering object system, which are not languages. We call the latter simply resources (R). We must note that all the things about which we speak in the languages Z dont belong into the R. Further we distinguish the languages between the languages used for the communication in the systemeering object system and between languages used to describe the models of the data production. The latter are rather norms or rules of data processing than means of communication and we call them old models (Z_{old}). The former group of languages we call professional languages (Z prof). Respectively a part of recources in systemeering object system is used for data production (R_{old}). In the process of systemeering we produce a metalinguistic expression of a professional language/languages, but only of the part of professional language, which we can formalize. So the professional language is divided into two parts; into the formalized object language (Z_o) and into other part of professional language ($Z_{\bar{o}}$). The result of the specification of object language is the metalinguistic expression of object language (Z_o^{Meta}): This is called LDSM in Klemola & Lyytinen. The part of objective reality potentially described by Z_o is called R_z respectively.

Systemeering (PJ & MK)

See def. of systemeering p. 13 in Klemola & Lyytinen

Systemeering as steps



$R = R_1 \cup R_{old}$

$R_1 = R_{1\bar{A}} \cup R_{1A} \cup R_{potential}$

$R_{1A} = R_{1\bar{B}} \cup R_{1B}$

$R_{potential} \supset R_{new}$

$Z = Z_{prof} \cup Z_{old\ models}$

Fig. 2. An outline of phases of systemeering

Systemeering (PJ & MK)

Z = languages

R = part of objective reality in the systemeering object system = resources

Z_{prof} = professional language (indep. of DP resources)

R_{old} = resources for old data production only

R_1 = other part of objective reality in the systemeering object system

$Z_{old\ models}$ = physical data system models in operation

Z_o = object language (a data system or a formalized part of the professional language)

R_z = part of objective reality described by Z_o (object system)

$\bar{Z}_o = Z_{prof} - Z_o$

Z_o^{Meta} = metalinquistic description of Z_o

$R_{potential}$ = new potential resources for data production

R_{1A} = resources in R_1 potentially feasible for new data production

$R_{1\bar{A}}$ = other resources in R_1

$Z_{resources}$ = description of $R_{potential}$ and R_{1A}

R_{new} = selected new resources for data production

R_{1B} = selected resources from R_{1A} for data production

$R_{1\bar{B}}$ = other resources of R_{1A}

$Z_{processors}$ = description of R_{new} and R_{1B}

Z_{models} = new physical data system models for operation

New traits: a) Z_{prof}, b) Z_o, c) consideration of messages in output reports as sentences of a certain language = object language $\subset Z_{prof}$

Some aspects: a) job design, b) management of systemeering
c) life cycle of data production system (Z_{prof}, Z_o, resources)
d) change analysis, e) PSC-systemeering model

On the other hand we must study and describe the resources, which can realize Z_o^{Meta} as a data production system ie. produce sentences in Z_o. We have potential recources outside the systemeering object system (R_{1A}) for the data production. These are described in $Z_{resources}$. As an outcome of systemeering we choose some established data production system or we take R_{1B} of potential resources R_{1A} and R_{new} of $R_{potential}$.

In the same time we produce a description or model of data production system (Z processors). This is called RM in Klemola & Lyytinen. The last phase in the systemeering is the production of models (Z_{models}), which express how the object language (described in the Z_o^{Meta}) is implemented as the physical sign system in the data production system (described in the Z processors). These models are called PDSM in the Klemola & Lyytinen.

3.2. Second day group

The major effort was to specify the relations between the term "systemeering" and other more established terms as "systems analysis and design" and "programming". In the discussion it turned out that the extension of the term "systemeering" includes all the activities, which belong into the extensions of the terms of "systems analysis and design" and "application programming". Referring to the PSC-systemeering model Pentti Kerola illustrated this discovery with the following picture (see fig. 3).

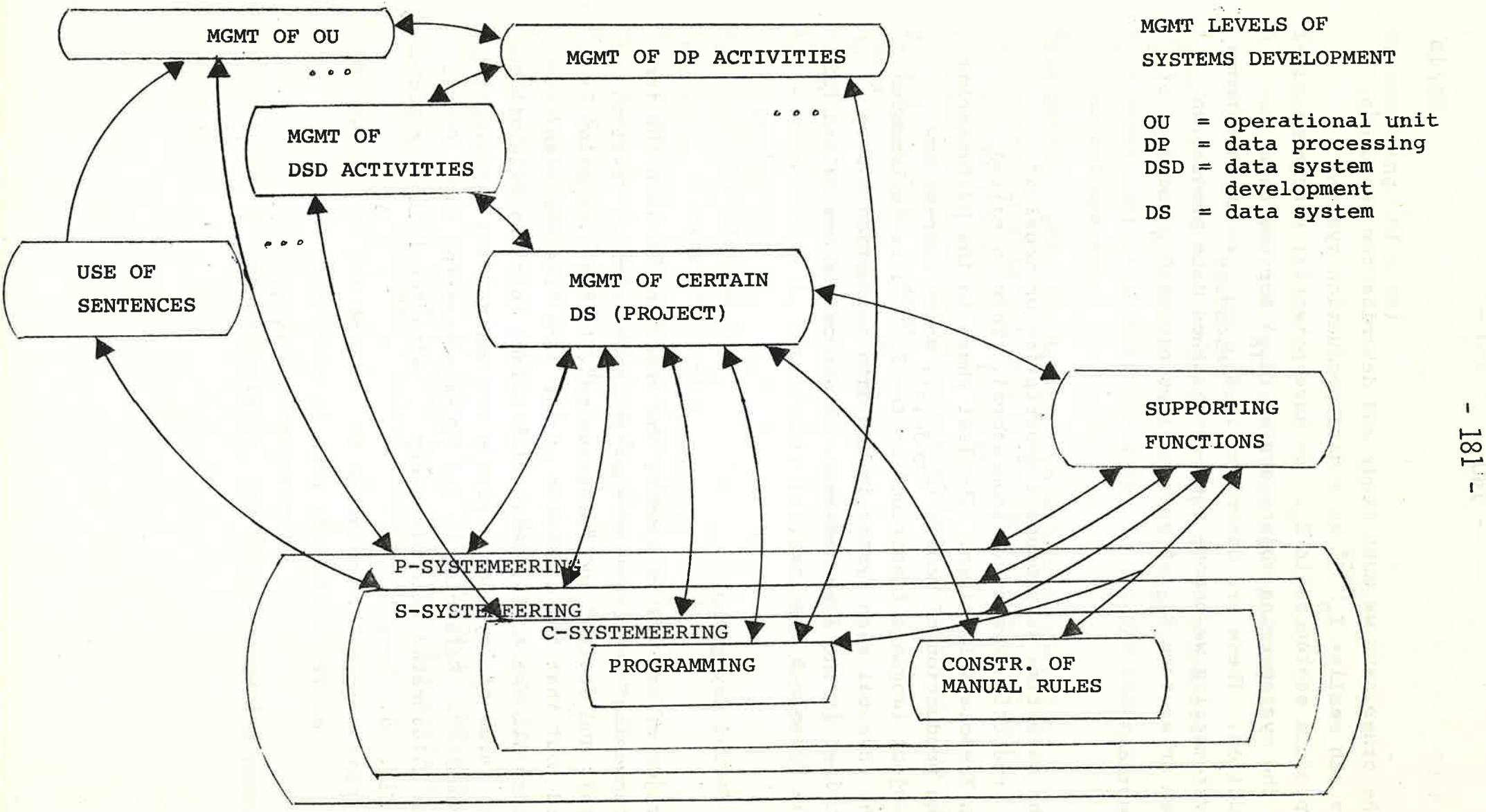

MGMT OF OU
MGMT OF DP ACTIVITIES
MGMT OF
DSD ACTIVITIES
USE OF
SENTENCES
MGMT OF CERTAIN
DS (PROJECT)
SUPPORTING
FUNCTIONS
P-SYSTEMEERING
S-SYSTEMEERING
C-SYSTEMEERING
PROGRAMMING
CONSTR. OF
MANUAL RULES
MGMT LEVELS OF
SYSTEMS DEVELOPMENT
OU = operational unit
DP = data processing
DSD = data system
development
DS = data system

Figure 3

The question if the programming belongs into the systemeering rose some discussion. At the first glance we can note, that if we study systemeering as an administrative whole the systems programming does not very often belong to the systemeering. Yet we need for example in the programming of real-time applications some systems programming skills and activities. Thus this kind of administrative view is not always very fruitful. The top-down view, for which it is obviously strived with the application of the term "systemeering", studies the programming as a part of systemeering.

One risk with the employment of the term systemeering is, that it is sometimes mixed with the term "systems engineering" Therefore it is recommended, where there is danger for confusion, to restrict the scope of term by the object of systemeering i.e. Data System Systemeering.

Main problem issuing the definition of systemeering concept is the wide range of knowledge - sciences - needed in the definition process. These sciences are far from eachother and their methodological presuppositions i.e. paradigm may considerably vary even be contradictory to eachother and inconsistent.

At the end of the group work the members of the group crystallized their opinions of systemeering in the following definitions:

Lyytinen
Systemeering of sentence production systems is a process of analyzing their effects and objectives, defining the sentence system, designing and testing the structure and activities of data production system implementing this sentence system in the limits of given or defined (during the process of systemeering) resources.

Järvinen
Data-systemeering is an activity where a sentence production system intended to serve certain purposes is designed and possibly implemented by potential dataprocessors. It can contain also a further study of service goals mentioned above.

Klemola

Systemeering means the definition of the sentence class of certain professional language that can be formalized (object language) and designing and performing the tasks, which are needed for the transforming these sentences into data production. In the different phases of the systemeering the given objectives and existing restrictions are taken into consideration.

Peltola

"Data production system (DPS) production" is an activity that starts from a task definition and ends with a (tested) data production system. It includes

- definition of the objectives of DPS;
- definition of the "data contents" (as a logical language) of DPS;
- definition of the production;
- processes ('data medium', 'data processors', algorithms or programs);
- testing of DPS;

As an iterative process and to the level in details as specified in the task definition. It does not include

- cange over of DPS (installation of DPS);
- maintenance of DPS.

Kerola

Systemeering is
on the one hand the definition of systemized information production for the different levels of decision makers regarding the objectives, influences, content, structure and operation and on the other hand the composition and testing the fulfillment of defined criteria. In essence the same as

App 1/Kerola-Järvinen

Systemeering is a hierarchical entity of tasks composed of the following actions: study, design, construction and test of operationality focusing on goals, influences, content, structure and operation of a data system (sentence production system).

4. CONCLUSIONS AND PERSPECTIVES FOR FURTHER WORK

In this section some conclusions about the work group on concept definition are drawn and perspectives for further work on this topic are outlined.

The most important hinder for getting very easily and softly valid concept definitions in any new area of research, such as systemeering, is the weak or unperfect reflection of the essence of the studied phenomena: its relationships with environment, structure and functioning.

Converging upon the real essence of the phenomena takes time and requires scientific, lively and open discussion on the subject of research. In the course of discussion various views and opinions can be amalgameted and the contradictories in the concepts can be resolved. This requires that the concept "milieu" of every participant taking part in this discussion must be adopted to other's conceptions of this subject. We can say that the process of concept definition is learning, self-learning and relearning at the same time. All this takes very much time and many discussions. In this view the results reported in this summary are only an opening of the scene: an attempt to be acquainted with the style of thinking of other researchers.

On the other hand every concept is valid in some concept structure (i.e. in relation to other concepts) only so long as this concept structure does not change substantially. This means that every concept is relatively valid and this validity depends

- on the use of this concept (c.g. purpose);
- on the evolution of the phenomena it generalises and reflects; and
- on the evolution of other concepts it is related with.

No concept is approved in scientific community by solely defining it, no matter how fine, logical or coherent the concept definition is, unless it is used in scientific thinking as a "work horse" in attempts to more deeply understand some phenomena. This means that these concept definitions must be tested in practice. Their scope of applicability must be studied etc. This must always be remembered when attempts are made to define some concepts by group work. The members of the group may close theirselves to fine, logical and closed world of pure definitions, which have a value in their own.

In the course of group work the members of group may "stick" to their own definitions and opinions. Therefore they can become blind for other's suggestions. At the same time they may try to dominate the whole discussion and in this way prevent many valuable insights from coming up.

One difficulty in the definition process is the lack of clear objectives for the definition process. For whom these definitions are denoted? When, where and how these concepts can be applied? These and other questions need careful study, if the concept definitions are hoped to be succesful.

Finally we will point out following essential conclusions about the usefulness of group work and its applicability to concept definition

- working was instructive and helped to understand other researcher's way of thinking ;
- the definitions given above must be applied in various contexts by the participants of the seminar. After this "field test" we can take a new round of discussion

- there is plenty of need for continuing working on the concepts because the workability of the concepts here outlined must be verified in the larger concept structure, where such concepts as "data structure" "information structure", "logical data structure", "physical data structure", "data organization", "data model", "data base", "file", "record", "logical record", "physical record", "data item" etc., etc. can be analyzed and defined.

IV COMMENT PAPERS

Comments on WG 2 report

Notes on terminology, abstraction levels and levels of detail in the PSC model of systemeering

Notes on the PSC model of systemeering

Condensed comments to some papers in connection with systemeering research seminar in Tampere 1978-08-21--24

How to define data system and information system

Notes concerning levels of detail and levels of abstraction

Timo Järvi

Turku 21.11.1978

Comments on WG 2 report

I comment only on sec. 2 of the report, because in sec. 3 even the WG itself did not reach an understanding.

I think that the set of definitions of the basic concepts is too complicated and by no way better than those already known (cf. e.g. Lundeberg - Andersen: Systemering - Informationsanalys, Studentlitteratur 74). It could be wise not to define any concepts without explicitely knowing where they are needed. So researchers should try to use old concepts and define new ones only when they cannot continue their work with old definitions or they see that new definitions have a clear advantage over the old ones. The time will show out which concepts and definitions will survive.

A more detailed critic to the definitions is presented in what follows:

A sentence has already a well known meaning. Usually nothing is said about the truth value of a sentence, it may be false, interrogative etc. Also in computer science sentence has an established definition which differs from the one of the WG.

Data is defined using "data medium". This is like defining "hen" as a female animal born out of hen's egg.

Because it is not always clear whether a user knows some "sentence" or not, so the definition of information is ambiquous. Although a person knows a "sentence", e.g. a complicated formula, he might look it up to make sure that he remembers it correctly. So this formula should be information for him.

The first definition states sentence system as the expression language where a meaning is associated to each word. The second one defines sentence system as an arbitrary subset of the expression language. Both of these definitions are rather hazy.

A data production system seems to be some automaton which can produce all "sentences" of a "sentence system".

A sentence production system is the same automaton with an end user who can start it and somehow control it.

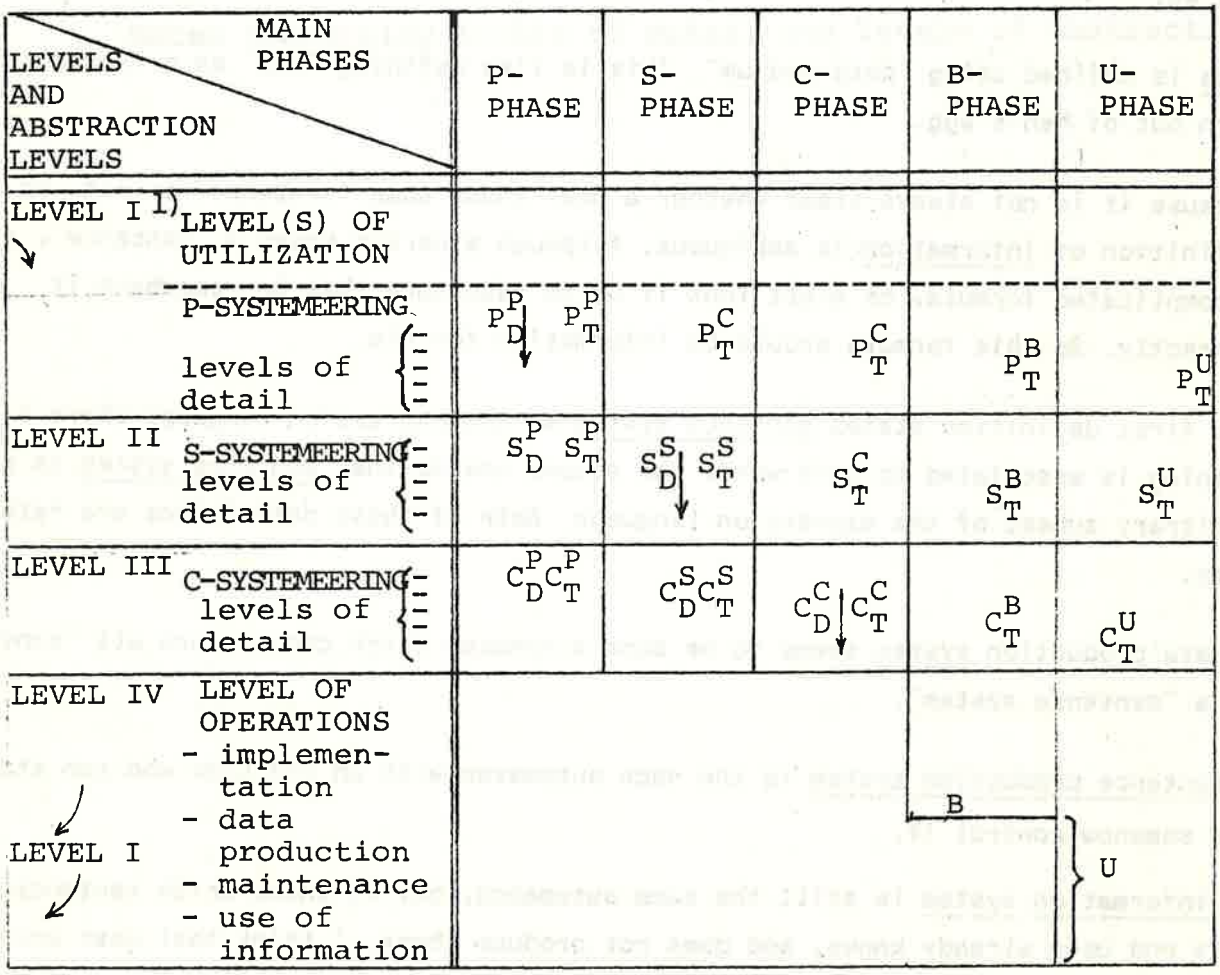
An information system is still the same automaton, but it knows which sentences it's end user already knows, and does not produce these. I think that most present information systems do not fulfill the definition given by the WG.

Pentti Kerola

NOTES ON TERMINOLOGY, ABSTRACTION LEVELS AND LEVELS OF DETAIL IN THE PSC MODEL OF SYSTEMEERING

This short comment paper tries to clarify some terminological ambiguity which has arisen from reports on the PSC model of systemeering. Such problems are very understandable and also useful for the future development of the model. The object of research concerned here is so complex and its theoretical basis so multidisciplinary that it is almost impossible to give an exhaustive description of the model in one paper or even in several papers. For historical and pedagogical reasons the "major ideas and basic principles" have been published first (and in Finnish, regrettably), and the next steps consist of testing, deepening, detailed explanations and naturally further development.

Let us illustrate the situation by the following figure.



1) See the circle with the levels.

The figure is a three dimensional description of the PSC model and its environment. The horizontal axis gives the main phases of the data processing system during its life-cycle. Vertically there are two dimensions:

1. Levels I-IV,[1] which contain the three abstraction levels:[2]
   - P-systemeering level
   - S- -"- -"-
   - C- -"- -"-

2. At each abstration level there are the levels of detail. The arrow marks the main phase, where most of the levels are designed and defined.

- - - -

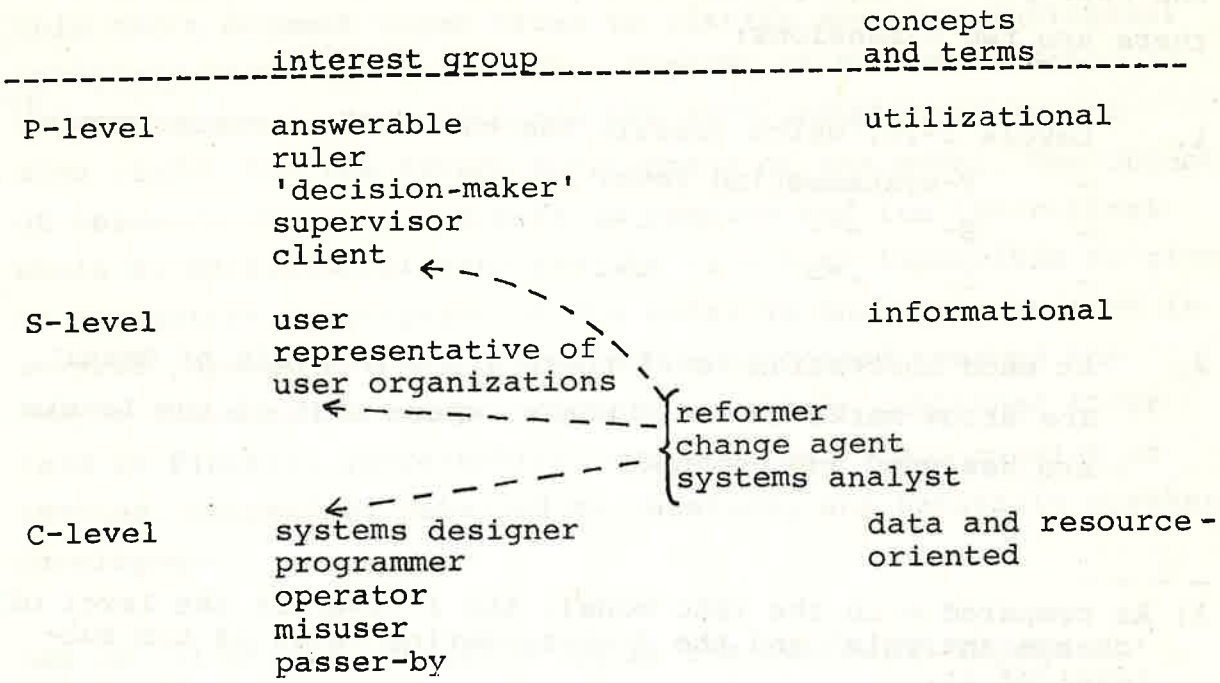
1) As compared with the ISAC model, the I level is the level of 'change analysis' and the P systemeering level is the sub-level of it.

2) WELKE has given very similar perspectives accordingly: (/1/)
   - SYSTELOGICAL perspective: How will changes to the existing information system alter/facilitate changes in the affected object systems (USS = user sub-system, DPS = data processing system (existing))
   - INFOLOGICAL perspective: How will changes to the existing information system alter/facilitate changes in the use of information (requirements, mode of presentation) by individuals who are members of the affected object systems?
   - DATALOGICAL perspective: How will changes to the information system alter/facilitate changes in the data processing sequences associated with the DPS-USS object system intersection.

   The initiating problem may arise from any of these perspectives. Each perspective, however, is valid once change is accepted, and each perspective then gives rise to its own set of problems (in the sense of either opportunities for improvement/correction if identified or consequences which diminish the potential effectiveness of the change if ignored).

   There is a natural ordering to these perspectives: systelogical, infological, datalogical. The greatest latitude for change exist at the systelogical level. Conversely, a particular systelogical solution (the composition and structure of the USS and DPS) constrains change at the infological and datalogical levels. At the infological level an individual must accept his/her position within the object system as prescribed by the associated systelogical solution. Within this position the need for various types of information (requirements) and their mode of presentation occur. These information requirements are the desired outputs of the datalogical solution, and hence constrain its solution.

Every abstraction level has its specific interest group (or decision-maker concerned) and the set of concepts and terminology necessary and suitable to it:



The table contains some typical terms describing the human groups concerned (see e.g. Welke 1977 /1/, DELTA-report 1974 /2/). The problematic situation concerns with the position of the reformer (or change agent or systems analyst). At which abstraction level do these work and with which set(s) of concepts? Historically we know that they have been and can potentially be put on every level. In that case we must also be sure that they really work at that level with the right knowledge background. Very often in practice the situation is that a systems analyst with knowledge derived from level C is designing the models for the other levels - with a conspicuous but unconscious lack of success. Today we look at the main objective of the systems analyst as being to catalyze the joint efforts of different people at different levels, if needed. His knowledge base must be broad and must overlap the abstraction levels. As Welke says:"Their task is to assist in the formation of logical and physical views of existing and proposed alternatives as well as identifying criteria for selecting an alternative".

On every abstraction level in the PSC model the major decision is made at the end of the main phase concerned. Then all the sub-tasks of the given main phase on the other abstraction levels will support the decision-making concerned. During systems development the levels of detail are developed separately on every abstraction level (note the contrast with the comment paper of Kämäräinen and also the useful deliberations of Tyllilä on the problems concerned). In conclusion the basis of the horizontal and vertical PSC partitions is the same, but apparently the terms selected are a little bit confusing and there are also proposals for changing them (see Mustonen in this report and Iivari 78 /3/).

Finally, we should make some comments about the second abstraction level, the semantic systemeering, and the S main phase. Here the basic elements of the concept structure are messages (combined or elementary ones) or sentences conveying potential information to the users. The problem of definition and selection of these information sets must be based on knowledge of the relations between the potential objects and the information sets representing them. The problem is a semantic one, and this is the principal reason for the selection of the term. Very often, and wrongly, we see that the main results of the information analysis are the sets on output and input data and the logical processing rules operating between them.[1)] In fact these are supporting results required in order to evaluate the informational results and the objectives and restrictions of the P level. Regrettably, this aspect is described vaguely in many systemeering models, or even not at all. In conclusion we could say that the term 'semantic' in the PSC model of systemeering is semantic in the wider sense compared with the same concept in semiotics.

---

1) note the difference, compared with the results of aspect C.

REFERENCES

1. R.J.Welke — Current information system analysis and design approaches: framework, overview, comments and conclusions for large-complex information system education. EDUCATION AND LARGE INFORMATION SYSTEMS, R.A.Buckingham (ed.), North Holland Publ. Co. 1977

2. R.Fjellheim et al — DELTA-rapport nr 2. Norsk Regnesentral, Oslo 1974

3. J.Iivari — Pragmatic control in the development of data processing function (in Finnish). A8, publ. by Institute of data processing science, Oulu univ., Oulu 1978

Heikki Kämäräinen

NOTES ON THE PSC MODEL OF SYSTEMEERING

The process of systemeering has been studied by Kerola and Järvinen /2.Kerola-Järvinen; 3.Leppänen-Kerola-Järvinen; see also preliminary papers in this report/. Their purpose has been to develop a set of thought constructs and to synthesize these into a systemeering model called PSC. Their work has brought clarity to the field of systemeering as well as provided a good basis for discussion.

The main construct of the PSC model is the set of P,S and C aspects. However, Kerola and Järvinen neither give a detailed account of the concepts that define these aspects nor suggest any concrete description techniques to document these three aspects of a data processing system. This makes it possible that the symbols P,S and C are used in several different senses.

These notes try
- to show that P,S and C aspects have two different meanings in the PSC model,
- to give a coherent interpretation of the PSC model, as well as
- to discuss in short an undesirable analogy drawn between the P,S and C aspects and a set of semiotic aspects.

1. Two different principles for multilevel systems

The classical subject-object setting provides a useful frame of reference for a discussion on levels. Let us assume that the subject is constructing a model of an object. This model can be single-level or multi-level. According to Mesarovic et al the basic reason for a multilevel model is the complexicity of the

object and the need for simplicity in the subject's model of the object. Mesarovic et al /5.Mesarovic et al, 37,63-65/ give also other motivation for multilevel models. The degree to which levels of a multilevel model depend on the features of the object as opposed to the subject's theoretical imagination can vary from one case to another.

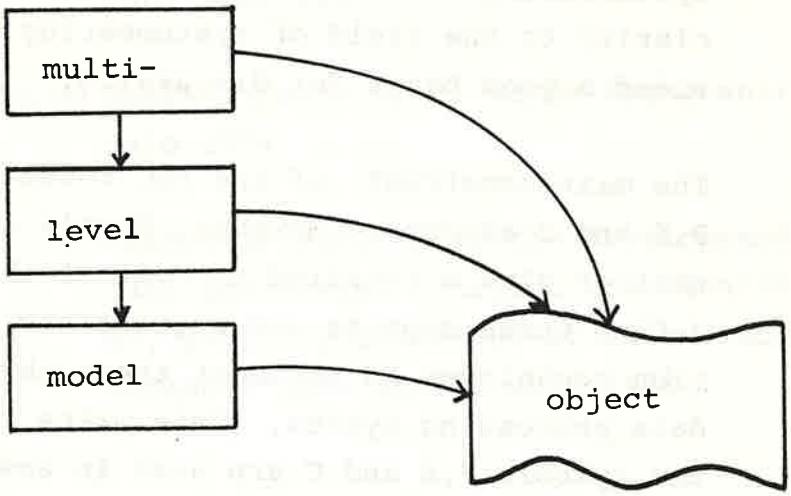


Fig. 1 An illustration of the subject's multilevel model of an object

During a systemeering process a systemeerer (subject) constructs models of a future real-world data processing system (object). These models may constitute a multilevel model of the data processing system. There are at least two different major principles that can be used to guide the construction of a multilevel model: the principle of levels of abstraction and the principle of levels of detail (these principles can also be combined in various ways).

The principle of levels of abstraction

By abstraction we mean that some of the numerous possible aspects of an object (in our case a data processing system) are taken out for inspection. On each level of abstraction the subject is concerned of a

definite set of object's aspects. These aspects are referred to by a set of concepts and by a corresponding language. On each level the object is modelled and described using the concepts and the language that are characteristic of that level of abstraction. A transition from one level of abstraction to another involves a transition from one set of concepts to another and from one language to another. The models from all levels of abstraction provide a many-sided description of the object.

The above characterization is an attempt to outline some essential features of the levels of abstraction. However, there are many kinds of objects that the subject may desire to model, and there are many ways in which the subject can perform the abstraction. Consequently, in modelling an object there can be many specific levels of abstraction and many specific principles of abstraction levels. Central questions in analyzing the levels of abstraction are

- what is the object that we abstract; what suggests the different levels of abstraction, i.e., the sets of concepts that define the levels;
- is there any relation that arranges the sets of concepts (which define the levels) into a hierarchy;
- may the sets of concepts (which define the levels) intersect. Kangassalo /1.Kangassalo/ gives a discussion on levels of abstraction in the context of developing a data base and a data processing system. He bases his hierarchical, non-intersecting levels of abstraction on the properties of data (i.e. on the object of manipulation in a data processing system).

The principle of levels of detail

When a view of the object (in our case, a data processing system) is fixed, it can be studied on several levels of detail. On each level of detail the whole object is covered by the corresponding model. A transition from one level of detail to another (i.e. the decomposition of the object) means that more details are unveiled, but it neither involves a change in the adopted view nor in the concepts or in the language that are used in modelling and describing the object. In other words, if we only change the level of detail, the concepts and the language that we use in the description of the object remain the same.

Mesarovic et al give an account on levels of abstraction in their book on hierarchical systems. Their concept of the level of abstraction is, however, not the one we mean here. It rather seems to be a combination of our level of abstraction and our level of detail. In their account it is clearly indicated that a transition from one level to another involves both a change of concepts /5.Mesarovic et al,41,79/ and a decomposition of the system /5.Mesarovic et al,42,84/. Some examples that are given clearly suggest this combined nature of abstraction levels /5.Mesarovic et al,30/ and some examples are not in contradiction with it /5.Mesarovic et al,38-40/.

2. An attempt to understand the PSC model

The PSC model (without implementation) is represented by the following symbol string:

(1) $P_D^P\ S_D^P\ C_D^P\ C_T^P\ S_T^P\ P_T^P \quad S_D^S\ C_D^S\ C_T^S\ S_T^S\ P_T^S \quad C_D^C\ C_T^C\ S_T^C\ P_T^C$.

The symbols P,S and C in the body of string (1) denote P,S and C systemeering; the symbols P,S and C as superscripts denote the main phases P,S and C of the systemeering process. The central construct of the PSC model is the hierarchical PSC structure $P_D\ S_D\ C_D\ C_T\ S_T\ P_T$. This hierarchical PSC structure is repeated in each main phase, although in modified forms. The representation (1) of the PSC model shows only the root of the hierarchy of systemeering tasks in each main phase. In every hierarchical PSC structure, except in the lowest level PSC structure, C systemeering involves decomposition of the object system into subsystems, and application of the hierarchical PSC structure to these subsystems. See ~~A4 of~~ this report and 2.Kerola-Järvinen.

A scrutiny of the definitions of P,S and C aspects /2.Kerola-Järvinen,15-19/, of their application in the decomposition of the object system, of the PSC model and of the illustrations of the PSC model gives rise to a suspicion that P,S and C aspects are used in two different senses in the PSC model.

The first meaning of the P,S and C aspects

According to the first interpretation, P,S and C aspects correspond to three levels of abstraction. In the PSC model the main phases represent levels of abstraction and thus the first interpretation of P,S and C aspects.

According to the P view (P level of abstraction) we
- concentrate on the concerned activity as a whole,
- study data processing activities amidst other activities,
- are interested in, whom the data processing activity serves and in what way.

According to the S view (S level of abstraction) we
- concentrate only on data processing activity,
- specify output data, input data, their component and precedence relations as well as the meaning of the data,
- study timing, volumes and frequencies of data.

According to the C view (C level of abstraction) we
- concentrate only on data processing activity,
- specify by what resources and methods the desired output data are produced from the specified input data according to their interrelations and their proper timing,
- study how to improve the efficiency of the data processing activity.

These crude definitions imply that the sets of concepts that define abstraction levels may intersect.

These abstraction levels are applicable only to data processing systems. The order of the abstraction levels is defined by pragmatic considerations; the levels of abstraction must aid in the design of a data processing system. Models of a data processing system on lower levels must be in harmony with models on higher levels of abstraction. On each level of abstraction the whole data processing system can be specified by the concepts that are characteristic of that level. On any level of abstraction the decomposition of the object system can be carried out to any desired level of detail. These levels of abstraction are global in the sense that they are not applied recursively to the subsystems of the object system; the potential decomposition of the object system takes place within each level of abstraction.

P,S and C levels of abstraction are probably based on a thought of three different types of decision-makers (managers responsible for the activity that uses the

output information, users of the output information, data processing specialists) that control the construction of a data processing system. The construction process is divided into three main phases corresponding to these three types of decision-makers and their areas of competence. Each type of a decision-maker approaches the data processing system from his own view and constructs a model of the data processing system using concepts that are in harmony with his interest and his area of competence.

Thus, when Kerola and Järvinen speak about P,S and C main aspects /2.Kerola-Järvinen,15-18/, P,S and C decision-makers and control /op.cit.,85,113/, P,S and C main phases /op.cit.,97-98/ as well as about P,S and C model(description) /op.cit.,109-110/, P,S and C should be interpreted according to the principle of abstraction levels.

The second meaning of P,S and C aspects

According to the second interpretation P,S and C aspects are auxiliary concepts which are used when decomposing a data processing system into subsystems. In other words, P,S and C aspects are used to organize a transition from one level of detail to another; thus P,S and C aspects in their second meaning are strictly related to the principle of levels of detail. In the PSC model the hierarchical organizations of P,S and C systemeering activities within the main phases (i.e. the hierarchical PSC structures) represent the second interpretation of P,S and C aspects.

In the following we outline using systems terminology the second meanings of P,S and C aspects. (Cf. 7.Sutherland,64-65).

According to the P view we
- analyze the environment of the system,
- specify, what environmental components the system affects and what the influences in these components are.

According to the S view we
- specify, what the system inputs and outputs are,
- specify, what the relations between inputs and outputs are (i.e. specify behaviour on macro level).

According to the C view we
- specify, what the system structure is on the next level of detail, i.e.
  - what the system components are (components: resources acting in their roles),
  - what the spatial distribution of the system components is,
  - what the relationships (division of labour, communication, control) between the system components are,
- specify, what the behaviour of the system components is (this step is applied only on the lowest level of detail).

We conclude that from the P and S views we see the system as a black box in its environment, whereas from the C view we see its internal structure on the next level of detail.

If it is desired, the P,S and C views in their second meaning can be considered as local levels of abstraction. They apply to any dynamic system. They are local, because they can be applied not only to a dynamic system but also recursively to its subsystems.

An interpretation of the PSC model

We have above sketched the two meanings of P,S and C aspects. In the following we try to give to the PSC model an interpretation that gives credit to these two meanings. See fig. 2.

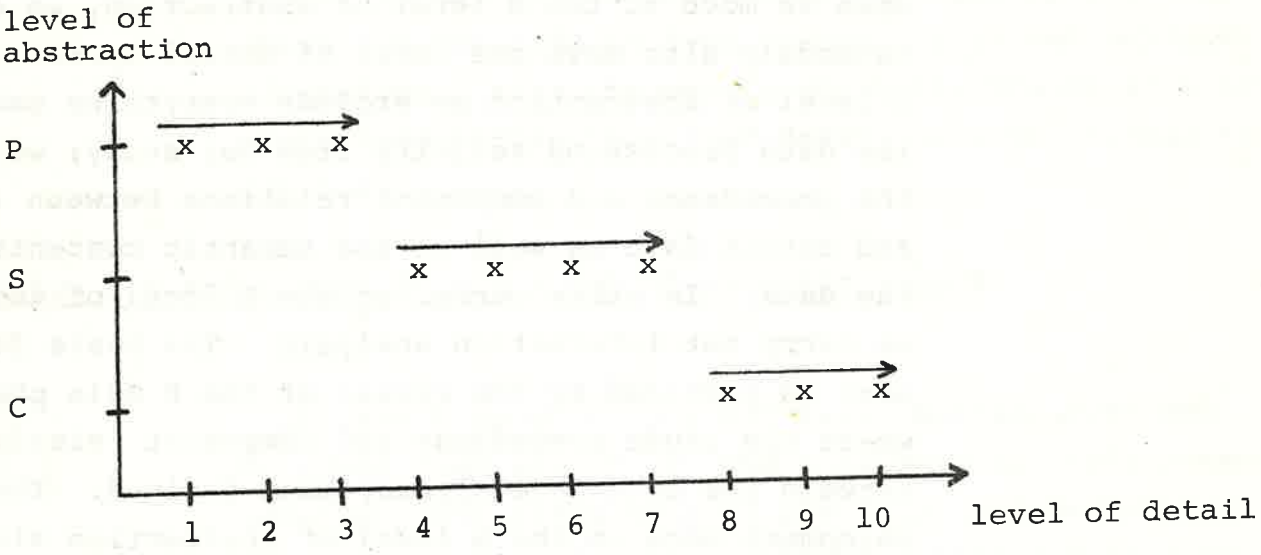


Fig. 2 The progress of systemeering process according to our interpretation of the PSC model

A systemeering process starts from the P level of abstraction (i.e. the P main phase). On the P level of abstraction we develop a model of the future activities of the object organization and of the future data processing activities amidst these activities. The development work starts from the highest level of detail. The modelling is aided by the P,S and C views (2nd interpretation):

(P) we specify the environment of the object activities;

(S) we specify the inputs and outputs of object activities as well as define the macro relation between inputs and outputs;

(C) we partition the object activities into sub-activities, i.e., we move to the second level of detail.

On the second level of detail we specify the sub-activities applying P,S and C aspects (2nd interpretation) to each of them. This brings us to the third level of detail. This refinement goes on until our conception of data processing activities in their organizational framework is detailed enough. Only then we move to the S level of abstraction (S main phase).

When we move to the S level of abstraction, we simultaneously also move one level of detail lower. On the S level of abstraction we exclude everything except the data processing activity from our study; we specify the precedence and component relations between input and output data as well as the semantic contents of the data. In other words, on the S level of abstraction we carry out information analysis. The basis for this work is provided by the result of the P main phase, where the crude precedence and component relations between the data have already been defined. The development work on the S level of abstraction also proceeds from higher levels of detail to lower levels and is aided by P,S and C aspects (2nd interpretation); (P) environmental analysis, (S) input-output relation, (C) subsystem structure.

When the results of information analysis are detailed enough, we move to the C level of abstraction (C main phase) as well as one level of detail lower. The purpose of the C main phase is to produce a model of data processing activities that is in harmony with the previous models as well as a model of necessary data processing resources. The basis for this work is provided by the result of the S main phase which identifies the potential processes as well as input and output data. Also on the C level of abstraction the development work proceeds from higher levels of detail to lower levels; the transitions are organized according to P,S and C aspects (2nd interpretation).

The above interpretation of the PSC model is based on the ISAC model of systemeering /4.Lundeberg et al/, and shows one way, how the principle of abstraction levels can reasonably be combined with the principle of levels of detail. Quite obviously, the above interpretation of the PSC model coincides in many respects with the ISAC model. The P main phase corresponds to change analysis and activity studies, the S main phase corresponds to information analysis, and finally the C main phase corresponds to data system design and equipment adaptation. Also the way in which the transitions take place from higher levels of detail to lower levels is similar.

3. The two P,S and C aspects and a set of semiotic aspects

P ,S and C aspects in their first meaning must not be identified with P ,S and C aspects in their second meaning. The range of applicability, the underlying thought models and consequently the meanings related to these two sets of aspects are different. The first set of aspects are aspects of a data processing system; they are based on the idea of abstraction levels and they can be applied only once to the object system. The second set of aspects are aspects of any dynamic system; they are based on the idea of decomposition and they are applied recursively when decomposing an object system into subsystems.

It is also worth noting that neither set of aspects should be identified with the pragmatic, semantic and syntactic aspects introduced by semiotics. The difference between the semiotic aspects as well as P , S and C aspects in their second meaning is discussed below at some length.

The P,S and C aspects in their second meaning are, as was stated, aspects of any dynamic system, whereas the semiotic aspects are aspects of language and semiosis A data processing system can be considered as a dynamic system that produces the sign component of semiosis, see 6.Nauta,27-28. Fig. 3 shows a black box model of a data processing system fitted together with a model of semiosis to form a model of a communication act. In the model of a data processing system we see input data x, behaviour f and output data y. In the model of semiosis /6.Nauta,27-28/ we see sign s (=output data y), interpreter i, effect e, denotatum or designatum d, and context c. Somebody or something generates the input data of our data processing system. The signs that form the input data refer to something d'. The data processing system manipulates input data according to a program f and produces output data. Output data refer to denotatum d. When the interpreter

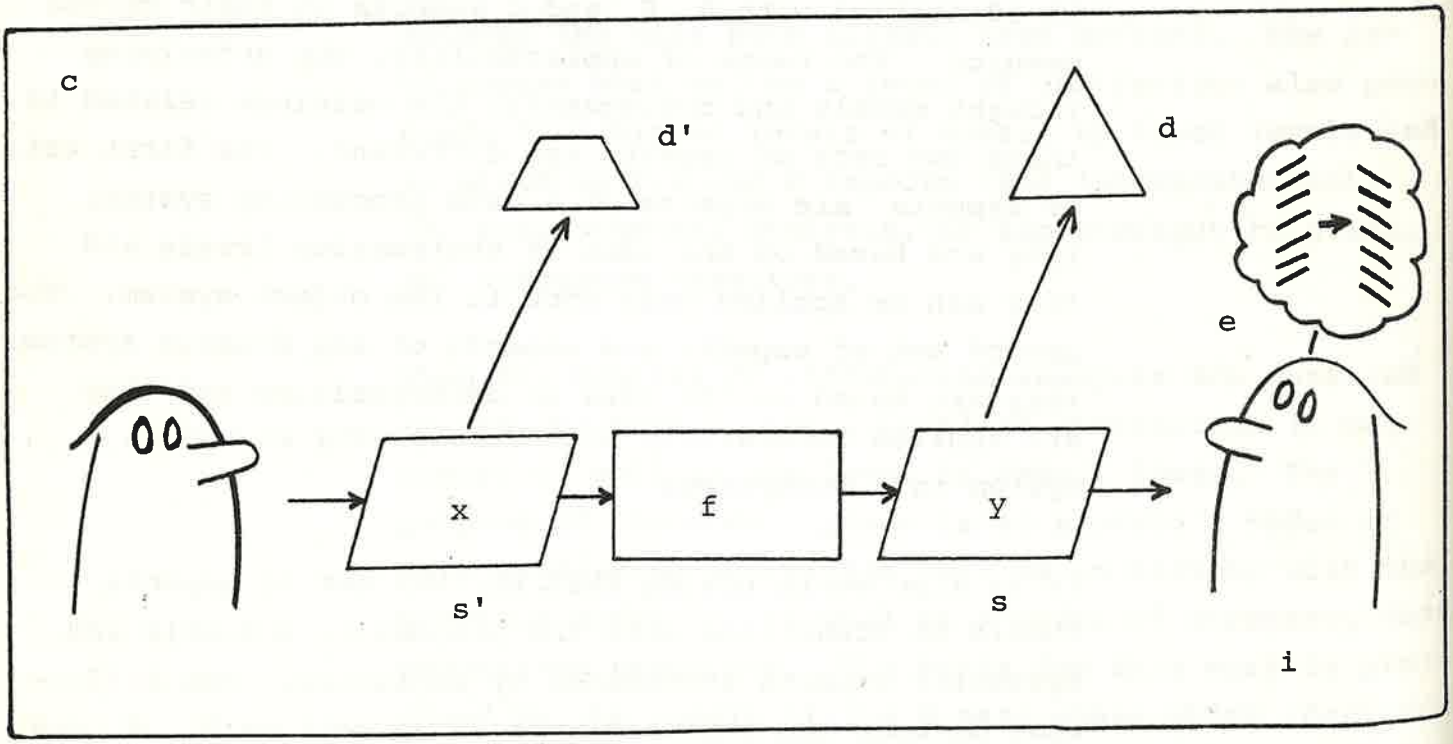


Fig. 3 A black box model of data processing system fitted together with a model of semiosis

receives the signs that form the output data, they have an effect on the interpreter, i.e., they call up a disposition in the interpreter to react in a certain way to denotatum d. This effect is sensitive to the context c where the interpretation takes place.

According to the P view we analyze the environment of the dynamic system and the influences of the system (in the form of output data) in the environmental components. According to the pragmatic view of semiotics we study the effect of the sign and the use made of it by the interpreter /6.Nauta,35/. As a data processing system is a sign producing system, the P view and the pragmatic view coincide in this case.

According to the S view we analyze the inputs, the outputs and their interrelations (i.e. macro behaviour). According to the semantic view we study the relationship between a sign and the conceptual entity (designatum) that it designates and/or the relationship between a sign and the physical entity (denotatum) that it refers to /6.Nauta,27-28,35,292/. These views are not the same, not even analogical. The former focuses on the precedence relation between input and output data, which is often a causal relation in a data processing system; whereas the latter focuses on the semantic/sigmatic relationship between data and their designatum/denotatum, which is a conventional relation created by a language community.

According to the C view we analyze the system structure on the next level of detail, i.e., we define the next level system components, their spatial distribution and their interrelationships. According to the syntactical view we study signs and rules relating signs /6.Nauta, 35,296/. These views do not coincide. The former view focuses on resources that produce signs: it focuses on the various data processing resources, their roles,

interrelationships between these roles, etc. . The latter view focuses on signs (input data, output data) and the combinations of signs that are formed according to the rules of the language to which the signs belong. It must be admitted that there is some analogy between these two views, as in both cases we are dealing with decomposition. However, the object to be decomposed is quite different.

4. Conclusions

It is possible that the above interpretation of the PSC model is based on a misunderstanding. Misinterpretations could, however, be avoided, if PSC model would

- be presented by symbols that do not give rise to undesirable identification of P,S and C aspects with other not dissimilar aspects,
- be presented by symbols that give credit to the two different meanings of P,S and C aspects,
- be presented in a fashion that clearly indicates, how the systemeering process proceeds from one level of abstraction to another and from one level of detail to another,
- be supplemented by a set of documentation techniques,
- be illustrated by an example where the documentation techniques are used.

REFERENCES

1. Kangassalo, Hannu — Conceptual structure of the file design problem: a general view. University of Tampere, Department of Computer Science, Difo report 8, Feb. 1975.

2. Kerola, Pentti; Järvinen, Pertti — Systemeering II. Gaudeamus, Helsinki, 1975. (in Finnish)

3. Leppänen, Mauri; Järvinen, Pertti; Kerola, Pentti — An introduction to data processing: user's view of data processing systems. Finnish Data Processing Association, publ. 37, 1977. (in Finnish)

4. Lundeberg, Mats; Goldkuhl, Göran; Nilsson, Anders — Systemeering. Studentlitteratur, Lund, 1978. (in Swedish)

5. Mesarovic, M.; Macko, D.; Takahara, Y. — Theory of Hierarchical, Multilevel, Systems. Academic Press, London, 1970.

6. Nauta, Doede — The Meaning of Information. Mouton, The Hague, 1972.

7. Sutherland, John — Systems. Analysis, Administration and Architecture. Van Nostrand Reinhold Company, New York, 1975.

isac
Mats Lundeberg
Göran Goldkuhl

COMMENTS 1(7)

1978-12-29

CONDENSED COMMENTS TO SOME PAPERS IN CONNECTION WITH SYSTEMEERING RESEARCH SEMINAR IN TAMPERE 1978-08-21--24

1 GENERAL COMMENTS

Many of the papers presented in connection with the seminar display interesting approaches within the research area. The semiotic starting point which is used in many of the papers is different from the dominating approaches within the area (e.g. the mathematically oriented relational theory). We have the impression that the semiotic starting point is broader in nature and therefore we think that this orientation will prove more fruitful in the future than many of the current approaches. However, we find a serious shortcoming in most of the approaches presented, the lack of empirical testing and empirical connections in general. Much of the research seems to be in an introductory state. It is therefore understandable that much effort has been devoted to the establishment of frames of reference instead of more empirically oriented research. We think that it is extremely important for the future work and results that enough effort is devoted to empirically oriented research. Our experience is that it is very rewarding for the theoretical work to be exposed closely to the problems in real life or practice.

We have found several links with our own research in the ISAC research group at the University of Stockholm. The PSC-model (Kerola-Järvinen) has great similarities with the systemeering model we are working with. We have got the impression that the current description of the PSC-model is of a conceptual nature. We doubt that all of the aspects can be covered in practical systemeering work.

The papers from Kangassalo, Kämäräinen, Klemola-Lyytinen and Nurminen present interesting semiotic frames of reference. We are looking forward to receiving and reading more concrete and empirically oriented research papers that build upon these frames of reference (which we thus find useful). This is also the case for the papers from Mustonen and Peltola--Maanavilja-Laine.

isac
Mats Lundeberg
Göran Goldkuhl

2 COMMENTS TO THE SPECIFIC PAPERS

2.1 Some comments to "An Introduction to the Basic Data Management Terminology" by Kangassalo

Kangassalo states "Between the level of data and the level of the real world there is a level of concepts". The purpose of his paper is among other things to describe some relations between the real world and data depicting it. We interpret his statement above graphically like this:

isac
Mats Lundeberg
Göran Goldkuhl

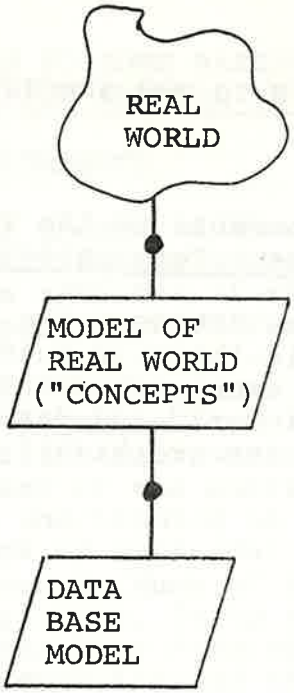


Figure 1 Our interpretation of Kangassalo

We would like to suggest the following alternative:

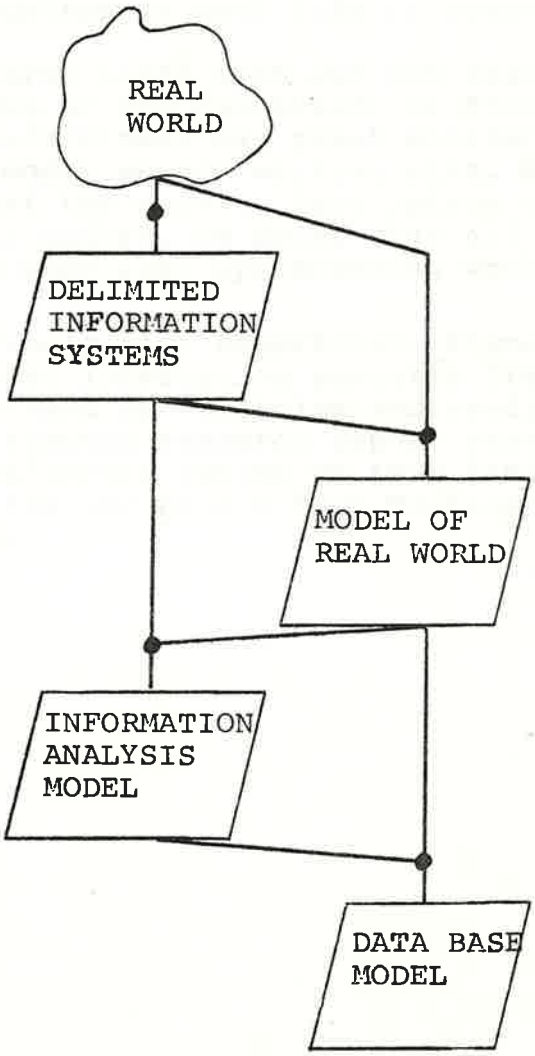


Figure 2 An alternative

isac
Mats Lundeberg
Göran Goldkuhl

As can be seen from the figures, our interpretation of Kangassalo is a subset of the alternative given in figure 2. The basic difference has to do with the sentence "Concept formulation requires a problem situation". The model of the real world can in our alternative be formulated with help of the delimited information systems. In the same manner, the data base model can be based on the information needs described in the information analysis model. Our alternative is thus "result"-oriented rather than "solution"-oriented.

2.2 Some comments to "PSC-Systemeering Model and Its Influence on Basic Concept Structure of Data System Development" by Kerola and Järvinen

1 There is a terminology problem with systemeering. In appendix 1, the authors state in the first sentence: "Systemeering has been defined as analysis and design of information systems". On page 4 the authors state "We found above that an implementation is lacking from our model of systemeering." Implementation is lacking because it is not a part of systemeering according to the definition above! We would therefore like to propose that sequence IV on page 5 is called PSC-information systems development and implementation model or something similar. By information systems development we mean analysis, design, and realization of information systems. Coding is part of the realization, whereas implementation is done in an organization when you start to use the information system. It is not clear to us whether the authors include realization in implementation.

2 We think that the authors conclusion from part II is extremely important: " The other conclusion ... is that it is not possible to give the fixed phase division model, applicable to every case of systemeering." According to our opinion, too much effort has been devoted to finding fixed phase models in e.g. Sweden.

2.3 Some comments to "An Analysis of Data Processing System and the Activities Related to It" by Kämäräinen

1 Our general impression of this paper is that it is thoroughly worked through analytically, but that it has not yet been applied. A natural next step would therefore be to apply Kämäräinen's analysis in practice.

2 In our impression of the paper, the first sentence in section 2 is a key sentence. Kämäräinen there states: "When thinking of and discussing data processing system it should (our underlining) ... be dissected into four different kinds of systems ..."

isac
Mats Lundeberg
Göran Goldkuhl

In our view, we use systems as an aid to look at the world. We choose how we look at it. Therefore we think that the sentence above should be rephrased: "When thinking of a data processing system it can be divided into ... The advantages of this are the following: ..."

We think that this is a key sentence for the paper, since the value of the analysis in the paper (although analytically well worked through) will not be clear until it has been demonstrated for what purposes it can be used in practice. Different analyses can be used for different purposes with different results - and this must be tested!

2.4 Some comments to "On the Conceptual Foundations of Systemeering - A Semiotic Approach" by Klemola and Lyytinen

Our comments to this paper are similar to the comments to the paper written by Kämäräinen, i.e.:

1 The authors present an interesting semiotic approach for developing a coherent systemeering theory. They state that they have only presented some general and necessary basic concepts for theory construction. The application of these concepts in practice will be an interesting next step in the research work of the authors.

2 The authors state on page 2 that the increase of theoretical knowledge of the nature systemeering is - in the long run - the best way to develop appropriate systemeering methods and tools. This may be so, but the authors do not include a discussion of how this theoretical knowledge shall be developed. We get the impression from statements such as these from page 12: "Any scientific theory serving practice has to explain its own existence, not only in terms of the conceptual starting points but in terms of the application of the theory. We are not very convinced that the latter task has often been successful apart from the former." that the authors maybe underestimate the advantages of developing theoretical knowledge by also including practical experiences. (Please note that this is not clear from the paper - it is more the impression we get.)

2.5 Some comments to "On Cybernetic Analysis as a Basis of Metatheories in Systemeering" by Mustonen

Mustonen's paper contains some of the main ideas and concepts from an earlier report. It is therefore not possible to give any detailed comments without this earlier report. As for some of the earlier papers, we feel that it will be interesting to see practical applications of the author's findings.

isac
Mats Lundeberg
Göran Goldkuhl

2.6 Some comments to "Properties and Problems of a General-Purpose Data System" by Nurminen

Nurminen gives a rather crude discussion about properties and problems of a general-purpose data system. We agree with his main points, e.g. that it is important to include the pragmatic and semantic points of view in both the definition of the data-base and its use. We also find the concept "domain of applicability" essential. Further comments to this work will have to wait until Nurminen continues with more detailed reports along the same line of thought.

2.7 Some comments to "Grammatical Data Base Model, I Description of Reality by Expressions of a Formal Language" by Peltola, Maanavilja, and Laine

The authors' aim is to form a method for representing the interesting part of the real world (the universe of discourse) by using a language. They call this linguistic representation of the real world a grammatical data base. Two important questions in this connection are:

1 How do you know what the universe of discourse is?

2 How do you know for what pragmatic purpose the grammatical data base is formed?

The authors state (page 7) that "We are, however, here describing the model and therefore pass over these aspects related to information analysis and hope to be able to discuss them at some other time." How can the authors be sure that the description of the model will not be affected by these questions?

3 SOME COMMENTS TO THE RESULTS FROM THE GROUP WORKS AT THE SEMINAR

The group work was performed around two different subjects:

WG1 Problems of performing research work in systemeering

WG2 Basic concepts of systemeering

We think that these two subjects illustrate the dilemma that systemeering researchers have. We think that most researchers agree that it is desirable to improve the basic concepts of systemeering. There are however different opinions of how these concepts should be developed.

isac
Mats Lundeberg
Göran Goldkuhl

One approach is to try and see concept formulation as a part of the research process as a whole (corresponding to subject G1 above) and another is to start working intensively with developing basic concepts (corresponding to subject G2 above). Since we believe in the first of these approaches, we think that the conclusions from the work on the second subject are important for future efforts in this area:

"No concept is approved in scientific community by solely defining it, no matter how fine, logical or coherent the concept definition is, unless it is used in scientific thinking as a "work horse" in attaining to more deeply understand some phenomena. This means that these concept definitions must be tested in practice. Their scope of applicability must be studied etc."

Another conclusion was that the process of concept definition is learning, self-learning and relearning at the same time. That is why we think that concepts must be developed within the context of an integrated view of the whole research process.

Kalle Lyytinen

Comment Paper
HOW TO DEFINE DATA SYSTEM AND INFORMATION SYSTEM

1. INTRODUCTION

The basic concepts needed for developing a theory of systeemeering must be consistent and conceptually well founded. In the papers presented and distributed to the participants of the seminar it is usual to denote with the same term to different things- so the terms have different senses - or it is denoted with the different terms to the same things. Furthermore it is usual to give various interpretations to the same term. For an analytic discussion on the terms (definitions) Data System(DS) and Information System(IS) I will present a preliminary schematization of the taxonomy of different proposed definitions.

The grouping(partition) of the given definitions is based on the basic concepts proposed in the paper by Mikko Klemola and Kalle Lyytinen. (cf p). At the same time I will show that all proposed definitions are logically deducible from the basic definitions given in that paper. In this sense we can say that these basic concepts are logically exhaustive for deriving all other proposed definitions of DS and IS.

2. THE FRAME STRUCTURE OF CATEGORIZATION

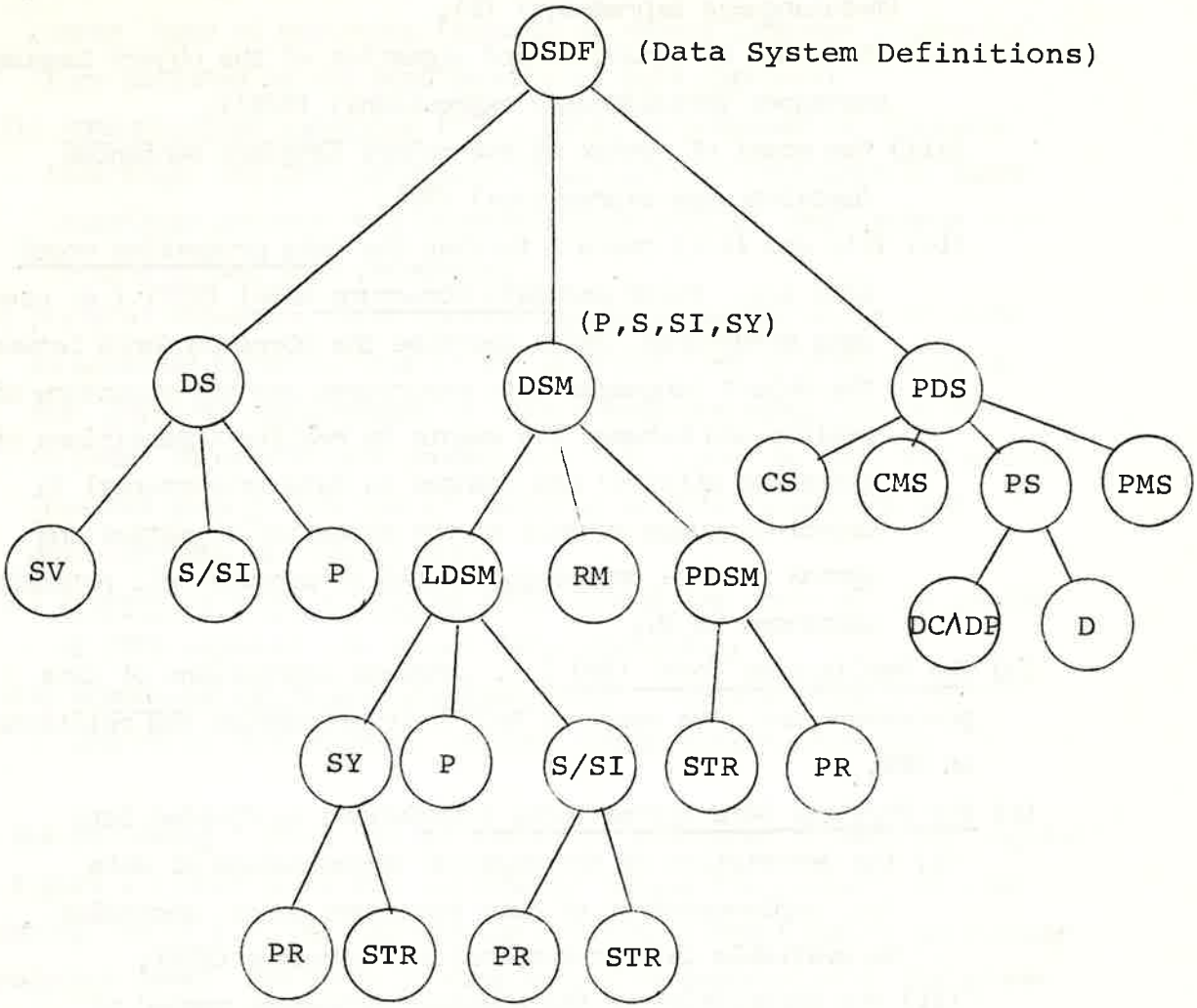
The most important resolution levels from the semiotic starting points connected with the definition of DS and/or IS are depicted in figure 1. These resolution levels as illustrated in figure 1 are an attempt to establish a basis for systematic classification for DS /IS definitions. For that purpose the partition of defining elements of DS/IS definitions must be exhaustive and the parts of each defining element must be pairwise disjoint or mutually exclusive. This partition of defining conceptual elements for DS/IS definitions will be called the frame structure of categorization.

The resolution levels in the frame structure can be classified into:

(1) object language level i.e. into level of Data System (DS);
(2) metalanguage level i.e. into Data System Models level (DSM);
(3) physical object level i.e. into physical objects which generate or control processes capable to handle physical objects as signs ,i.e. into Data Production System level (DPS).

(1) is divided according to semiotics into
   (a) pragmatics of object language (P);
   (b) semantics and sigmatics of object language (S/SI)
   (c) syntax of object language (SY).

Figure 1 Resolution levels and the frame structure for examination of definitions of data system and information system



(2) is divided into as we propose in our paper (klemola&Lyytinen,1978)

(a) The Logical Data System Model (LDSM) where we have the following components

(i) the model of pragmatics of the object language sentences (metalanguage expression) (P);

(ii) the model of semantics and sigmatics of the object language sentences (metalanguage expressions) (S/SI);

(iii) the model of syntax of the object language sentences (metalanguage expressions) (SY);

(iv) (ii) and (iii) contain further the data processing model (PR) i.e. rules and data structure model (STR) i.e. used data structures, which describe the correspondence between the object languages data structures and the structure of reality and between the events in reality (transitions of states of affairs) and changes in data (structures) in object language as well as the semantics,sigmatics and syntax of data processing needed to produce all potential sentences in DS;

(b) The Realization Model (RM) i.e. language expressions of data processors and data carriers and their properties and relationships in DPS;

(c) The Physical Data System Model (PDSM)which is divided into

(i) the description of the physical organization of data i.e. implementation of data structure model according to available data processors/data carriers (STR);

(ii) the description of the physical processes needed to carry out the processing model (PR);

The PDSM is considered particularly as a language expression which has a representation form suitable for human comprehension.

Because all models on the second level (2) are metalanguage expressions they have as linquistic structures their own pragmatics(P),semantics(S), sigmatics (SI) and syntax (SY).

(3) level contains physical objects. It is consisted of

(a) physical objects, which create physical processes needed to manipulate signs. These objects are described in RM.

We can decompose these obejcts into

(i) processing subsystem (PS) which is consisted of on the other hand data processors (DP) and of data carriers (DC) and on the other hand of sentences (data D) of object language in physical form dictated by the reguirements of data carriers;

(ii) communication subsystem (CS) , which is composed of physical interfaces and data processors/data carriers contained in these interfaces between data processor/data carrier combinations performing actual data processing

(b) physical objects , which control/determine manipulating of signs as physical objects. This is the physical form of PDMS suitable for physical requirements of data processors specified in RM in order to create data production. This - what we call Implemented Physical Data System Model residing in data processors and data carriers is divided into

(i) processing model subsystem (PMS) containing parts of STR and PR of PDMS relevant to actual data processing;

(ii) communication model subsystem (CMS) containing parts of STR and PR of PDMS relevant to data communication processes.

In the following I shall use abbreviations connected to each rectangular in figure 1 (junction of row and column) inside the parentheses when denoting to it i.e. the particular defining element inside particular resolution level. The corresponding column is depicted with the abbreviation outside the parnetheses i.e. this identifies the resolution level. Furthermore we can join to examinations of metalanguage expression level prefixes P/S/SI/SY before the resolution level expression. These prefixes illustrate the corresponding semiotic levels, if we want to show on what semiotic level we are investigating these metalanguage expressions (models).

Examples

(a) DPS = examination of data production system in general

(b) DS(P) = pragmatics of object language i.e. pragmatics of data system

(c) DSM[LDSM(P∧S/SI(PR)] = pragmatic descriptions and semantic-sigmatic processing descriptions in metalanguage in LDSM of DSM. If there is no danger of confusion we can leave the prefix DSM away, as we shall do in the following pages

(d) S(LDSM) = the semantics of LDSM. In this case we can see the prefix showing the semiotic level of examination, because we are considering metalanguage expressions.

3. CLASSIFICATION OF PROPOSED DEFINITIONS

Information System Def (1) IS= DS(P)∧ DPS

- definition proposed by Lundeberg &Goldkuhl
-implicitly in the way term is used by Kerola&Järvinen

Information System Def (2) IS = LDSM(S/SI)
- implicitly in the way term was used by Nurminen

Data System Def (1) DS = DS(P)∧ DPS

- pragmatic definition of DS given by Kerola&Järvinen

Data System Def (2) DS =DS(SY)∧LDSM(SY(PR))

- semantic definition of DS given by Kerola&Järvinen

Data System Def (3) - DS = DPS

- cf the definition of the Data production system in the summary of the group work 2

Data System Def (4) = DPS(PS(D)∧ PMS∧DS(S/SI∧SY)

-cf Kämäräinen's definition of system of data and processing

Data System Def (5) DS = DPS(PS(DP∧DC)∧CS∧CMS

-cf Kämäräinen's definition of DS as a system of resources

Data System Def (6) DS = S(LDSM∧RM∧PDSM)

-cf Kämäräinen's definition of DS as a system of models

Data System Def(7) DS = DSM

-cf Kämäräinen's definition of DS as a system of descriptions

4. CONCLUDING REMARKS

The definitions analyzed above show a great variety of starting points for defining the concepts of DS and IS. We can say, that the essential point is not what term you use in the definition of concept but on what conceptual basis your definitions (definiens) rest. Or to put it another way, how consistent and exhaustive is your concept structure used in your definition.

I think that all the definitions given above are reducible (as was shown) into to the basic concepts given in semiotic approach. The same is not true in the other direction.

Pekka Tyllilä

NOTES CONCERNING LEVELS OF DETAIL AND LEVELS OF ABSTRACTION

1. Introduction

In this paper the principle of levels of detail and the principle of levels of abstraction are studied. The aim is to show some problems in the semantics of these concepts. A comparison is made of the use of these concepts in some reports. In that connection my own interpretation of the meanings of these concepts is given.

This kind of analysis is important, because it shows an ambiguity in the principal meta concepts of systemeering. It is hoped that this analysis would provide a suitable basis for the conceptual analysis of systemeering models e.g. the PSC model of systemeering.

2. The principle of levels of detail

This principle is generally used in analysis and design of systems /1, p. 60/2, p. 18/3, p. 27/. It is a tool for partitioning an object of study to more easily conceivable parts. It also helps us to analyze, what the conceivable parts of the whole are and how they form the whole. In Kämäräinen's "Notes on the PSC model of systemeering" we can find a characterization of this principle: "When a view of the object (in our case a data processing system) is fixed, it can be studied on several levels of detail. On each level of detail the whole object is covered by the corresponding model. A transition from one level of detail to another (i.e. the decomposition of the object) means that more details are unveiled, but it neither involves a change in the adopted view or in the concepts nor in the language that are used in modelling and describing the object." /4/

The principle of levels of detail is a special case of problem representation by problem reduction. This is an analytical truth, if we accept the interpretation that the view within which the levels of detail are generated is defined by the problem statement or more exactly by the problem definition. (Note that the concepts of a "view" and a "definition of a view" can be found in some papers of systems analysis".) The different levels of detail are the different levels of subproblems. The complexity of the objects of study (the complexity of subproblems) is the dimension which defines the different levels of detail. /5/

Let us make a closer analysis of problem representation by problem reduction. It is a good tool in solving complex problems. It aids us in managing the complexity of a problem by partitioning the original problem into subproblems, and if they are still too complex to be solved, the subproblems into subsubproblems etc. It is natural to require that the solutions of the subproblems on every level provide the solution of the original problem.

The principle of problem reduction suggests that the problem is solved by partitioning it into subproblems /5, p. 3-44/. However, this principle does not help us, in practice, to do the problem reduction or to find the solution of the lowest level subproblems, as it does not indicate the order in which the different subproblems and their solutions are generated. We have two basically different ways to generate the problem tree: breadth first and depth first, see fig. 1 /6, p. 2/. Usually the factual problem trees are compromises between these two principles.

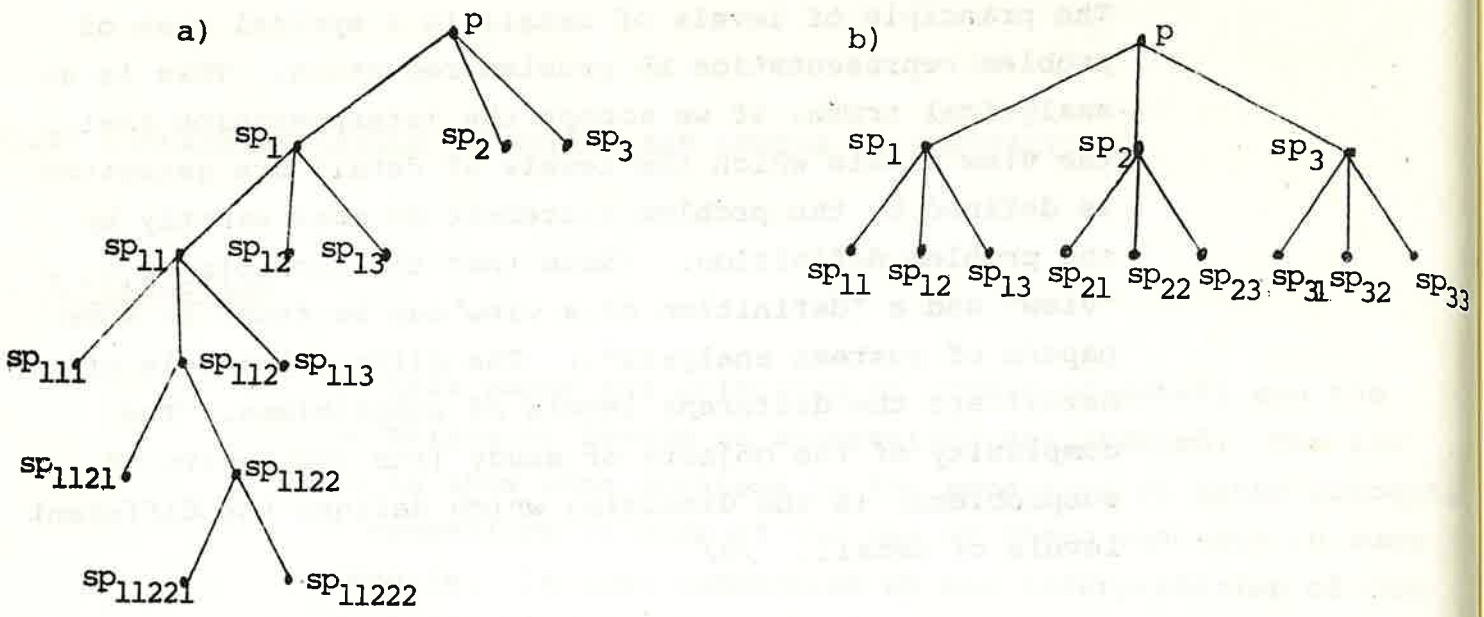


Figure 1. Illustration of the depth first (a) and breadth first principles

The conclusion of our analysis concerns the application of the principle of levels of detail. This principle does not provide complete guidance to the flow of systems work, as it does not indicate the order, in which the subsystems are developed. In addition to problem reduction principle there is another general method for problem solving, called the state-space representation of the problem, which provides better directing hints how to proceed. In this sense the state-space representation of the problem would do much better. The state-space representation is defined as follows:

A path $s_0 \overset{f_1}{\rightarrow} s_1 \overset{f_2}{\rightarrow} s_2 \overset{f_3}{\rightarrow} \ldots \overset{f_i}{\rightarrow} g$ should be found such that $s_0 \in S_0$, $s_1, s_2, \ldots \in S$, $g \in G$ and $f_1, f_2, f_3, \ldots, f_i \in F$ where S_0 is the set of initial states, S is the set of all possible states, G is the set of goal states and F is the set of transition functions. /5, p. 43/

3. The principle of levels of abstraction

The principle of levels of abstraction is also called the principle of levels of description or the principle of stratified description /7/. Let us analyze the meaning(s) of this concept by answering the following questions:

1) What is the meaning of the term "level"?
2) What is the meaning of the terms "abstraction" and "abstract"?

Intuitively we have some ideas of the "right" use of the concept (term) "level". It is correct English and conceptually right to speak about levels of intelligence or about levels of detail. In these cases it is easy to see the dimension that defines the different levels. It is fundamental for the intuitively correct use of the concept "levels of something" to express the dimension or the classification that defines these levels. - It is possible to formalize this. In an n-dimensional space one of the dimensions defines the levels, and the rest of the dimensions (n-1) form the levels. It is, perhaps, not necessary that each level is defined by one point of the defining dimension. In some cases a level is defined by a set of points; in these cases it is possible to talk about the depth of the level /8, p. 9/.

In the following we analyze some meanings of the terms "abstract" and "abstraction". "Abstract" is an adjective. As an adjective, it refers to some property. The property can be measured and compared with other similar properties: something is more or less abstract. Abstraction is a process, the result of which is something that is abstract. So far we have explained "the syntactical semantics" of the terms (concepts) "abstract" and "abstraction". But what is the deeper semantics of these concepts in our case? We quote some dictionaries:

"Abstract is something that is separated from, what is real or concrete; thought of separately from facts, objects or particular examples /9/."
"Abstract is something that is conceived apart from matter and from special cases; it is theoretical, not applied /10/".

"Abstraction is an act of considering something as a general object apart from special circumstances /10/."

These are the partial meanings of concepts "abstract" and "abstraction" that are needed in our discussion. Based on these definitions it is possible to specify that

- abstraction is a process of generalization; the objects of this process are usually concepts or thoughts;
- abstract is an adjective that measures the results of the process of abstraction.

Now we are ready to discuss the meanings of "the principle of stratified description" and "the principle of levels of abstraction". In the sense defined by Mesarovic, Macko and Takahara these concepts are identical. This is a consequence of the fact that the concepts "abstract" and "abstraction" are defined in a loose enough way. These loose definitions allow the formulation of five principles, which characterize the principle of stratified description /7, p. 40-42/.

Kangassalo has developed the concept "principle of levels of abstraction". At first glance it is not clear, to what extent Kangassalo's concept is identical with the stratified description of Mesarovic et al. Kangassalo has made more specifications. The object that he studies is data and the objects of abstraction are the properties

of data. Kangassalo's thesis is that it is possible to order the abstracted properties of data in such a way that the least abstract properties are on the lowest level and the most abstract properties are on the highest level. The degree of abstraction is defined here as a property of the concept that denotes the property of data. "The degree of abstraction of a concept can in principle be measured by counting the number of levels in the hierarchy of concepts used to define that concept." Figure 2 illustrates Kangassalo's levels of abstraction of data /8/.

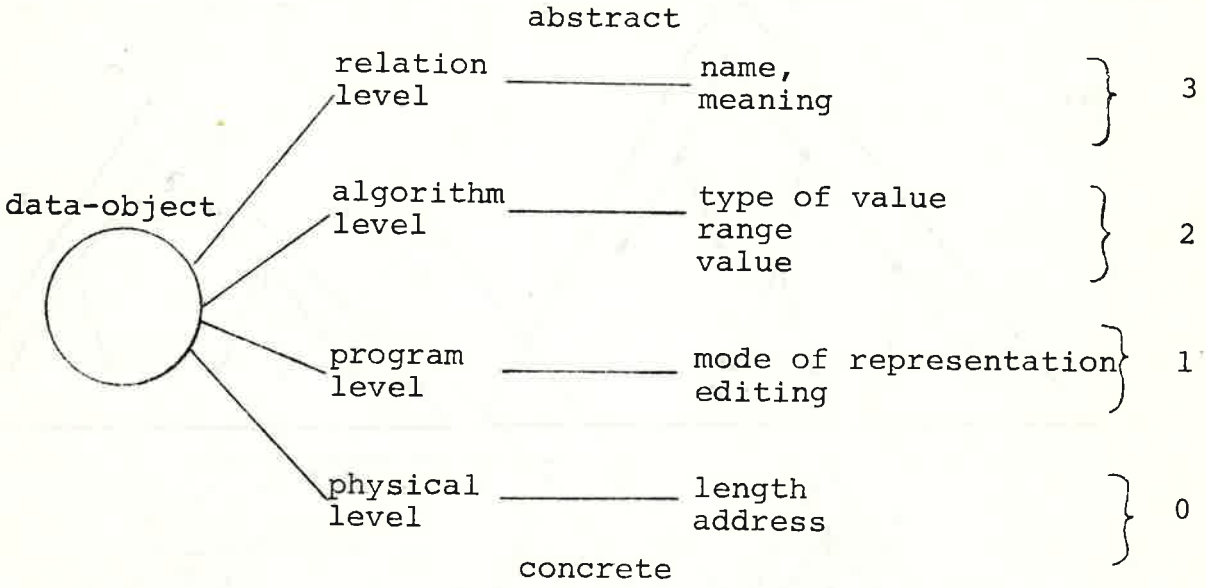
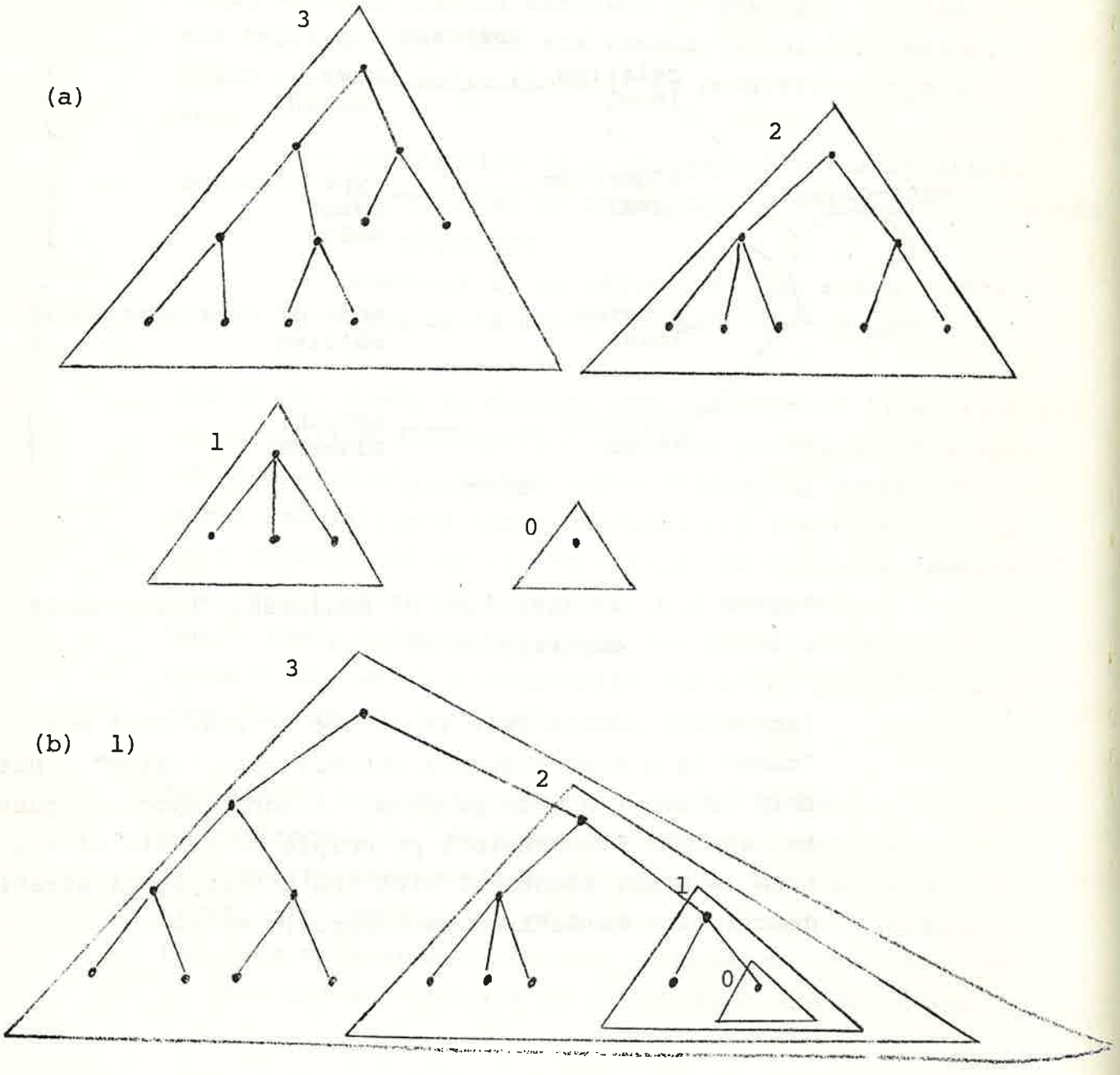
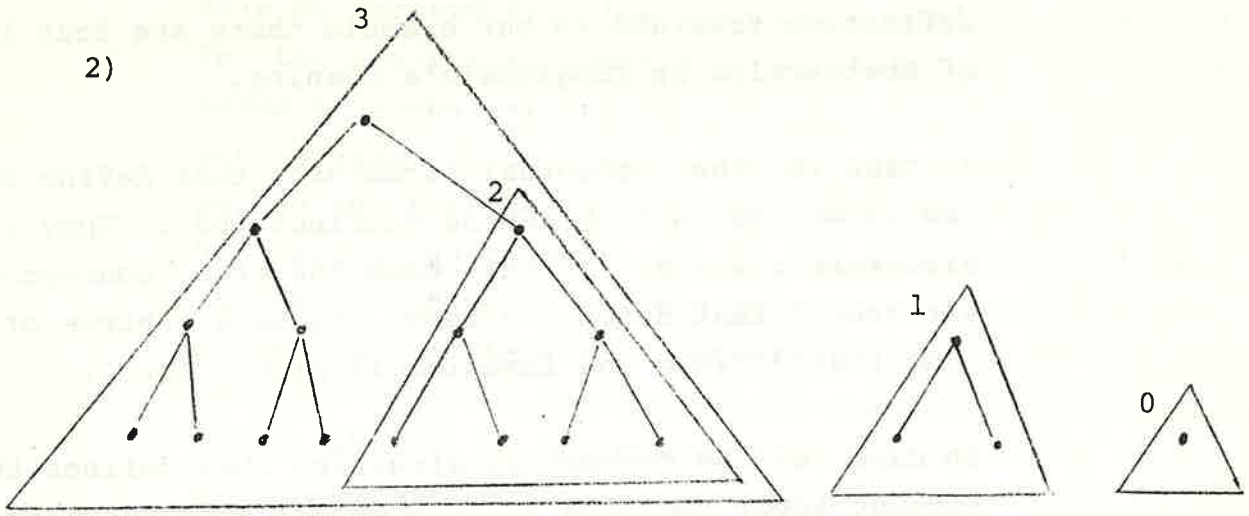


Figure 2. Illustration of Kangassalo's levels of abstraction of data

Kangassalo admits that it is not trivial that e.g. "name" is a more abstract concept than "value". But this is not the main problem. A more important question is, whether Kangassalo's principle of levels of abstraction is still identical with the principle of stratified description as defined by Mesarovic et al..

From Kangassalo's "principle of levels of abstraction" it is possible to infer three different types of conceptual structures that define the levels of abstraction. Figure 3 illustrates these three types of conceptual structures; nodes stand for concepts and edges stand for relations between definiendum and definiends. Triangles (polygons) and numbers represent different possible levels of abstraction.





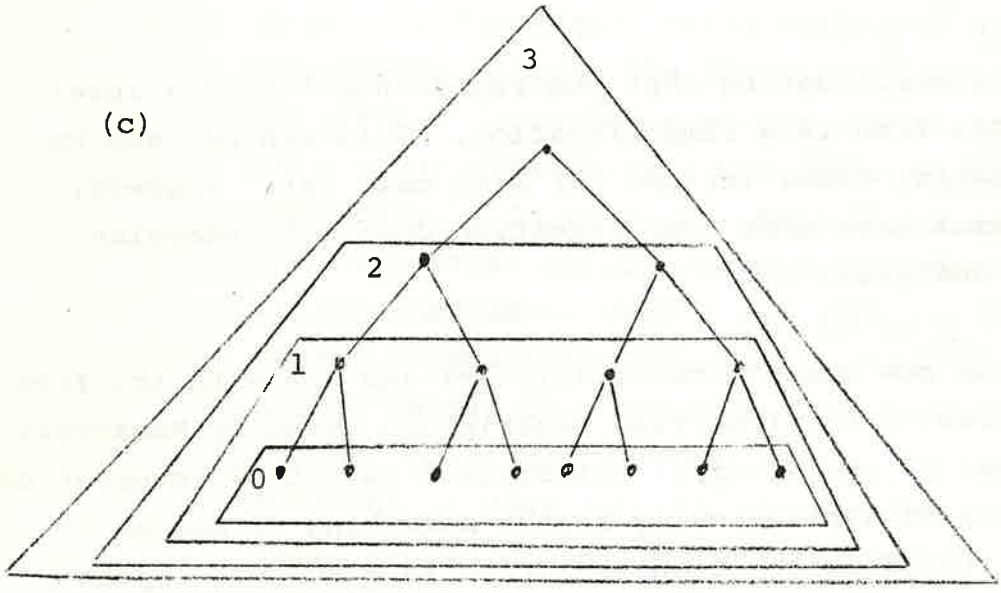


Figure 3. Illustration of the different possible conceptual structures that can be inferred from Kangassalo's principle of levels of abstraction.

In case (a) we have four distinct conceptual structures. They are arranged with respect to the depth of the definition tree and in our example there are four levels of abstraction in Kangassalo's meaning.

In case (b) the conceptual structures that define the levels of abstraction can be distinct (b2). They can also form a whole. In that case the tree (conceptual structure) that defines a level may be a subtree of the tree that defines the immediately higher level.

In case (c) the conceptual structure that defines the highest level includes the conceptual structures that define all other levels. The conceptual structure that defines the second highest level includes all other levels except the conceptual structure that defines the highest level, etc.

It is worth noting that the representation of a level by one tree is a simplification. This can be seen by comparing cases (a) and (b) with case (c). However, I think that this simplification does not undermine our analysis.

Let us now compare cases (a), (b) and (c) with the five principles of stratified description given by Mesarovic et al. /7, p. 40-42/. Principle 1 is not of interest to us, as it synthetically states something of how a stratified description is formed in a subject object relationship, whereas we are dealing with the logical (analytical) properties of stratified description. Principle 5 states: "Understanding of a system increases by crossing the strata: in moving down the hierarchy, one obtains a more detailed explanation, while in moving up the hierarchy, one obtains a deeper understanding of its significance /7, p. 42/. Similarly it is not of interest to us, as it is a synthetical statement.

Principle 2 states: "Contexts in which the operation of a system on different strata is described are not, in

general, mutually related: the principles or laws used to characterize the system an any stratum cannot generally be derived from the principles used on other strata." In the analytical meaning it is clear that this principle holds in cases (a) and (b) (it is a truth by definition), but in case (c) it does not hold. In Kangassalo's paper we can see that he has noticed this /11, p. 50-51/. In his definition of different levels he does not allow structures of type (c). More generally, in a principle of abstraction levels of Kangassalo type it is impossible to allow type (c) conceptual structure, if the principle should also be in accordance with the principle of stratified description: there is a contradiction between conceptual structures of type (c) and the second principle of stratified description. From now on, only cases (a) and (b) are studied.

Principle 3 states: "There exists an asymmetrical interdependence between the functioning of a system on different strata. ...; for a proper functioning of the system on a given stratum, all the lower strata have to function correctly." In case (b1) this is an analytical truth: the conceptual structures defining different levels are interconnected in such a way that the requirement of correct functioning on lower levels in relation to higher levels is a property of a model of functioning. Whether principle 3 holds or not, is not an analytical question in case (a). It is a synthetical (empirical) question concerning the object system; it is not a property of the (conceptual) model.

Principle 4 states: "Each stratum has its own set of terms, concepts, and principles. What is an object on a given stratum becomes a relation on a lower stratum; what is an element becomes a set; a subsystem on a given stratum is a system on the stratum below. ... On any given stratum one studies the behaviour of the respective

systems in terms of their internal operation and evolution, while the study of how these systems interact so as to form a higher stratum is studied on that higher stratum. This is important since it indicates that the studies on the lower strata are not necessarily better, more fundamental, or more basic than those on higher strata. On the lower strata one concentrates on the operation of the subsystems, having the study of their interrelationships for the higher strata. This would not be so if on the lower strata one considers the entire system just as on the higher strata; this, however, is usually not the case, since the principles and methodology on any stratum in general are not suitable for this procedure."

This characterization of principle 4 gives rise to a few notes. What is actually the deeper meaning of a level's "own set of terms, concepts, and principles", if the relationship between different levels is so reducible that "what is an object on a given stratum becomes a relation on lower stratum ...". The conclusion "that the studies on lower strata are not necessarily better, more fundamental or more basic than those on higher strata" is also very strange, because it is impossible to study the connections between the parts of an object without knowing the behaviour of the parts.

There seems to be an ambiguity in principle 4. The underlined principle is not the same as the explanation below it. In one meaning the explanation suggests an almost type (c) conceptual structure. Only the concept of connection, the way in which the lower level concepts are connected to form higher level concepts, does not match type (c) conceptual structures. I think that this is a wide question about analytical and synthetical statements, because the concept of connection is logical in this case.

The underlined part of principle 4 is an analytical truth in case (a). However, it is only partially possible to infer this principle from the conceptual structure (b).

The conclusion of our analysis is that, if type (b) of conceptual structures are adopted then principle 4 of stratified description is only partially true; to principle 3 type (b) of conceptual structures give the best match. The converse holds for conceptual structures of the type (a): conceptual structures of type (a) match principle 4 very well but give a poor match to principle 3.

4. Conclusion

On the basis of our analysis it should be clear that there are and it is possible to form different principles of levels of abstraction. The relationships between these different principles are problematic. This leads into difficulties in the interpretation of these principles.

I would also like to draw attention to the question of extensionally defined concepts as a basis of theory. Our hope is, that this paper would show the importance of extensionally defined (meta-)theoretical concepts, if we want to discuss problems of a theory or to develop a theory.

REFERENCES

1. Kerola, Pentti
Järvinen, Pertti
Systemeering II. Gaudeamus, Helsinki, 1975. (in Finnish)

2. Jahnukainen, Miikka
A frame method for planning an information system for the firm. The Finnish Economic Association, Helsinki, 1971. (in Finnish)

3. Leppänen, Mauri
Järvinen, Pertti
Kerola, Pentti
An introduction to data processing: user's view of data processing systems. Finnish Data Processing Association, publ. 37, 1977. (in Finnish)

4. Kämäräinen, Heikki
Notes on the PSC model of systemeering. See comment papers of this report.

5. Eloranta, Kari, T.
Heuristiikat ja heuristisuus. Käsittelyongelmista ja niiden ratkaisemisen metodologiasta hallinto-opin näkökulmasta. University of Tampere, Tampere, 1974. (in Finnish)

6. Kurki-Suonio, Reino
Notes on some tree-search algorithms. University of Tampere, Department of Computer Science, Report A3, Tampere, 1973.

7. Mesarovic, M., D.
Macko, D.
Takahara, Y.
Theory of Hierarchical Multilevel, Systems. Academic Press, New York, 1970.

8. Kangassalo, Hannu
Conceptual structure of the file design problem: the general view, Difo Report 8, University of Tampere, 1975.

9. Hornby, A.,S.
Oxford Advanced Learner's Dictionary of Current English. Oxford University Press, Oxford, 1977.

10. Barnhart, C.,L. (ed.)
The American College Dictionary. Random House, New York, 1967.

11. Kangassalo, Hannu Stratified Description as a Frame Model of File Description. DIFO Year Book 2/1973, pp. 47-67, Tampere, 1973. (in Finnish)

V FUTURE PLANS

All the participants rated the seminar very highly as having been useful and encouraging for future work. The following preliminary plan of meetings has been accepted:

1979 Seminar, May-June, Turku
1980 Summer School, Stockholm (Åland)
1981 Seminar, May-June, Oulu

Potential participants will mainly be decided by reference to their research interests.